REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	10 Nov 97	Final Report

4. TITLE AND SUBTITLE

Enhancement, Decomposition, And Wavelet-Based Compression Of Space Images

5. FUNDING NUMBERS

F6170896W0305

6. AUTHOR(S)

Dr. Pavel Chochia

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Institute for Information Transmission Problems
19, Bolshoy Karenty
Moscow 101447
Russia

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

EOARD
PSC 802 BOX 14
FPO 09499-0200

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

SPC 96-4085

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

A

13. ABSTRACT (Maximum 200 words)

This report results from a contract tasking Institute for Information Transmission Problems as follows: The contractor will perform a service consisting of investigation in enhancement, decomposition, and wavelet-based compression of space images. He will research and develop effective nonlinear methods of image enhancement and restoration; construct and demonstrate a wavelet-based image compression algorithm.

14. SUBJECT TERMS

Imaging, Physics

15. NUMBER OF PAGES

36

16. PRICE CODE

N/A

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Institute for Information Transmission Problems
Russian Academy of Sciences (IPPI RAN)

European Office Of Aerospace Research
And Development (EOARD)

Enhancement, Decomposition, and Wavelet-Based Compression of Space Images

Final report

P.A.Chochia, O.P.Milukova, and V.B.Tkhor

Principal Investigator: Chochia Pavel Antonovitch, Ph.D.
Mailing Address: IPPI RAN, 19, Bolshoy Karetny, Moscow, 101447, Russia
Phone +7 (095) 209-4781 Fax: +7 (095) 209-0579
E-mail: chochia@ippi.ac.msk.su

Business Office: Institute for Information Transmission Problems
Russian Academy of Sciences (IPPI RAN)
Address: 19, Bolshoy Karetny, Moscow, 101447, Russia
Tel. +7(095) 209-4981 Fax: +7(095) 209-0579

Signature: P. Chochia *Date:* 10.11.1997

INTRODUCTION

Video data, coming from satellites, space probes, air and space labs, appears to be one of the most significant information channel in remote sensing, aerospace researches, monitoring of Earth resources. The registered images often turn out to be distorted due to non-ideal characteristics of capturing devices and measuring conditions. The volume of data exceeds the transmitting channel bandwidth, and finally the receiver often desire to extract only some interesting information from incoming data. Thus the important part of this problem is image processing, including image restoration, enhancement, and compression.

The report contains the results of researching and developing effective image enhancement and restoration methods, and of constructing powerful wavelet-based image compression algorithm. An approach to decision of the problems is based on proposed multiscale image model, which describes an image as a combination of statistically and semantically independent components, containing details of different sizes and different information classes. Statistic distinctions of these components allow to find effective algorithms for decomposition of received image to several information components. Discussing about *image decomposition* we mean the splitting of an image onto the set of such components. It offers the possibility to take from an image only the component of interest, to avoid redundunt information from the following analysis, and to create decomposition-based image processing methods. Basing on this approach, algorithms for image enhancement, restoration, and compression are developed and considered in the project. In particular, algorithms for image smoothing, local contrasting, automatic amplitude correction, blurred image restoration, decomposition-based image compression.

Speaking about image enhancement and image restoration we always use some criteria for evaluating of image quality. Usually such evaluation criteria are also qualitative and informal – the estimate of visual image quality is closely connected with the human perception [1]. Typical estimates, analogous to mean square deviation, are enough far from the visual quality, moreover, they are applicable only to a couple of some images, not to a single one. Variations may be more adequate and helpful estimates. We would like to apply to the task of image estimate one little-known two-dimensional variation, proposed by Kronrod (published in [2]). This estimate is based on notation an image as some continual surface over two-dimensional space of coordinates. In discussing formal measures we rely on the *two-dimensional variation*, which is based of continual image model.

Discussing about image measures we will consider basic concepts of the continual image modeling, assuming any discrete model as some approach to the continual one. On this base we will discuss metric characteristics on an image set, the question of image smoothness, and two-dimensional image variation. Continual model will be used for the definitions and explanations of it. For the developing of image decomposition and enhancement algorithms, we will use discrete model. Also the discrete analog of two-dimensional variation will be used for experimental estimating of image smoothness.

The report contains three following chapters: i) mathematical image models, ii) image decomposition and enhancement, and iii) wavelet-based image coding. In models' domain the discrete models for small local vicinity and large image fragment are proposed, which are later used for constructing different image processing algorithms. Continual image model is used for discussing about some new image measure – two-dimensional variation, and as the appropriate basis for image restoration algorithm. Image decomposition is treated as some specific image smoothing, and obtained results are later used for image enhancement. Algorithm for automatic gray scale image correction is constructed basing on of local image vicinity model. The task of restorating images, distorted by convolution-like operator, is decided on the basis on continual image model. In final chapter the task of image encoding, using preliminary image decomposition and wavelet basis, is considered.

1. MATHEMATICAL IMAGE MODELS

All the numerous mathematical models for image processing can be divided into two classes: class of continual, and class of discrete models [3]. Continual models are much more suitable for the analytical research, they help to perform theoretical investigations and estimates, but actually they describe some ideal objects, which we cannot operate with. Discrete models much more correspond to real digital images, because in a computer we may contain and process only discrete signal. Such models give us direct way to develop image processing algorithms, but bring difficulties in theoretical analysis. So nor continual nor discrete model is unable to solve all the problems, and we will try to apply both models for our task.

1.1. DISCRETE IMAGE MODEL

1.1.1. Basic approaches

Image decomposition may be interpreted as the task of splitting an image to the smoothed and the difference components. The smoothed component should contain lengthy image areas with approximately constant intensity, separated by sharp edges. Nonlinear integral operators [8, 9] and algorithms, based on anisotropic diffusion models [10] are used for the decision of the problem. In our work we will take into account the nonlinear algorithm.

Typical observed scene consists of a set of objects of different sizes [11], so the image of the scene will include corresponding set of areas (domains) of smoothly changed brightness, separated by sharp contour edges. Each area contains fine details and texture, and often is distorted by some noise. We assume that captured image $x_{mn} \in \mathbf{X}$ may be represented as an additive mixture of several statistically independent components, each next one is characterized by reduced scale parameters.:

$$x_{mn} = s_{mn} + t_{mn} + u_{mn} + \dots \ .$$

First component s_{mn} determines smoothed values of extended domains and describes extended image details. The other's terms reflect information about fine details, texture, noise, etc. In a simply case one may limit the row of components by two [9]:

$$x_{mn} = s_{mn} + t_{mn}. \qquad (1.1)$$

Here t_{mn} becomes to be the sum of all other information components.

Let an image $\mathbf{X}$ consists of R different domains $U^1,\dots,U^R$, and inside a domain U^r ($x_{ij} \in U^r$) smoothed component S with appropriate accuracy is represented by polynomial of power ϖ. Then one may write the formula:

$$S_{ij}^r = \sum_{p=0}^{\varpi}\sum_{q=0}^{p} a_{pq}^r i^{p-q} j^q .$$

For the value of individual element we obtain:

$$x_{ij}^r = \sum_{p=0}^{\varpi}\sum_{q=0}^{p} a_{pq}^r i^{p-q} j^q + t_{ij}^r, \qquad (x_{ij}^r \in U^r). \qquad (1.2)$$

This is the main formula of image model. Analysis of real images demonstrates, that inside unique region we may assume t_{mn}^r to be normally distributed $N(0,\sigma_r^2)$, but with different variation σ, because statistical properties of texture component t_{mn}^r may differ from a domain to domain.

Basic formula (1.2) may be modified with different ω. Depending upon the class of processed images, tasks to be decided, required accuracy, and computing complexity one can choose appropriate polynomial power value ω. Our experiments demonstrate, that for the majority of real airspace images and typical window sizes (10-30 pixels), the objects inside the window are usually specified by slowly changing or approximately constant brightness. Because of this reason, and due to the necessity to develop enough effective algorithms for image enhancement, it is demanded to restrict the power of polynomial as $\omega = 0$.

Almost all image models describe statistical correlation between neighbor elements only inside a small vicinity (3-5 pixels). However statistical dependencies between pixels in unique domain lie much far – up to the distances of 15-20 elements and larger [11]. This leads us to use different models for the small vicinity and for the extended image fragment.

1.1.2. Model of local fragment

Let W_{mn} be a fragment, surrounding central point (m,n), and covering J domains $U^1,\ldots,U^J$. Correspondingly to (1.2), and under $\omega = 0$, the elements of fragment inside domain U^r are described by piecewise constant model:

$$x_{ij}^r = s_{mn}^r + \tau_{ij}^r, \qquad (x_{ij}^r \in U^r) \tag{1.3}$$

where s_{mn}^r is an average brightness of part of domain U^r, falling into the fragment W_{mn}. This model is simple, enough close to the majority of real images, and acceptable for the decision of the following task of image decomposition.

1.1.3. Model of local vicinity

Let us consider a vicinity V_{mn} of some shape around image pixel x_{mn}, containing R elements $x_{mn}^r \in V_{mn}$ $(r = 1,\ldots,R)$, and ρ^r denotes the distance between x_{mn} and x_{mn}^r. Draw a plane minimizing mean square error, which will form a two-sided angle of value ψ_{mn}. Thus to each point with its vicinity we may assign a vector $\mathbf{g}_{mn}$ with amplitude $g_{mn} = \mathrm{th}\,\psi_{mn}$ and rotation angle ν_{mn}. Let elements of vicinity differ from this plane by random value γ_{mn}^r. In such an event one can write following formula for the model of vicinity, which correspond to (1.2) under $\omega = 1$:

$$x_{mn}^r = \mu_{mn} + \rho^r g_{mn}^r + \gamma_{mn}^r, \tag{1.4}$$

where μ_{mn} is the value of the plane at central point (m,n), and g_{mn}^r - the projection of the vector $\mathbf{g}_{mn}$ to the vector, drawn from center of vicinity to point r.

The task to draw such a plane is not difficult and similar question was considered in [3]. Let α and β are tangents of plane slopes in vertical and horizontal directions, i and j - relative coordinates of the pixel in vicinity. In such a case the equation of the plane will be $z_{ij} = \mu + \alpha i + \beta j$, and formula (1.4) will be written as $x_{ij} = \mu + \alpha i + \beta j + \gamma$. Mean square error will be given as

$$f(\alpha,\beta,\gamma) = \sum_i \sum_j (\alpha i + \beta j + \mu - x_{ij})^2 .$$

Minimum of $f(\alpha,\beta,\gamma)$ locates at a point, where all partial derivatives equal to zero, so one can find values of α, β, and μ. In practice the vicinity of 3x3 pixels is most frequently used. For such a case:

$$\alpha = \frac{1}{6}\left(\sum_j x_{1,j} - \sum_j x_{-1,j}\right); \quad \beta = \frac{1}{6}\left(\sum_i x_{i,1} - \sum_i x_{i,-1}\right); \quad \mu = \frac{1}{9}\sum_i \sum_j x_{ij} . \tag{1.5}$$

Knowing α and β we can simply calculate parameters of vector $\mathbf{g}$:

$$g = \sqrt{\alpha^2 + \beta^2}; \qquad \nu = \mathrm{arctg}\left(\frac{\beta}{\alpha}\right). \tag{1.6}$$

1.1.4. Local average and local median

In accordance to (1), the decision of image decomposition task consists of defining the value of smoothed component s_{mn} for each image pixel. We will find decomposition algorithm in the class, based on analysis of image brightness distribution for local fragment, rounding current processed elements.

The subtask of image decomposition - finding smoothed image component - is similar, but is not covered by local image smoothing task, performed or with the helps of local average

$$y_{mn} = \frac{1}{(2L+1)^2} \sum_{i=m-L}^{m+L} \sum_{j=n-L}^{n+L} x_{ij} , \tag{1.7}$$

or with the helps of local $\sum_{k=0}^{med\{W\}} h_w(k) = 1/2$ median

$$y_{mn} = med\{W_{mn}\}, \text{ where } med\{W\} \text{ is the root of the equation: } \sum_{k=0}^{med\{W\}} h^W(k) = 1/2. \quad (1.8)$$

Here $\mathbf{H}_{mn}^W = \{h^W(k)\}$ is the fragment histogram, i.e. the fraction of pixels of value k in the fragment W_{mn}.

1.2. CONTINUAL IMAGE MODEL

1.2.1. Definition of an image set

According to the continual model, any monochromatic, flat, and stationary image is represented by a bounded real-valued function of two variables $f(x,y)$; $(x,y) \in D$. A real-validity and a boundedness are the only restriction to $f(x,y)$ placed in image modeling. It is obvious that a set of such functions turns out to be too wide. In particular, it contains various "bad" functions to which many mathematical operations are not applied. To avoid such kind of problems it is supposed, that there exist all the properties of function, we need for one or another mathematical method. So we think that the function $f(x,y)$ is as good, as it is necessary. The result of this assumption turns out to be very unexpected, enough to mention the difficulties we have under using the Fourier transform for image processing problems.

Here we will try to formalize the properties of images, to underline which mathematical object we are dealing with, under considering an image a signal that can be percepted by human visual system. We will regard the simplest image type (flat, stationary, and monochromatic) as the electromagnetic radiation distribution with the density $f(x,y)$, which is defined on the bounded domain $D \in OXY$.

Earlier it was mentioned that function of two real variables $f(x,y)$ is a nonnegative, bounded, and real-valued, since it is proportional to squared amplitude of electromagnetic wave. The second obvious image property is the integrability of $f(x,y)$ on $D \in OXY$. Really, every physical quantity is a function of a domain, not a point. Like an energy, for example. On the other hand, a density is a function of a point, since a density is introduced as a derivative. So by definition $f(x,y)$ is a integrable function on D, and because of its boundedness on closed bounded domain, integrable function $f(x,y)$ is continuous on D everywhere except a null set.

Now we will obtain the third property of an image, rose from the physical model we use. To stay within the borders of geometrical optics, we have to consider only plane electromagnetic wave inside every small domain of a space. This means that the amplitude of the wave is a constant or low-changing function at the distances compared with the wave length λ or the resolution of a recorder [7]. This means that we can consider the light amplitude and the image $f(x,y)$ to be piecewise constant or piecewise monotone function.

Thus, it is shown that an image $f(x,y)$; $(x,y) \in D$ comply with following conditions:

1. $0 \le f(x,y) \le C$, where C is a constant;
2. $f(x,y)$ is a continuous on D, everywhere except a null set;
3. $f(x,y)$; $(x,y) \in D$ is a piecewise-monotone function.

So a set of images is a set of real-valued functions of two variables $f(x,y)$; $(x,y) \in D$ comply with the conditions 1-3 above. These properties can be used to solve many image processing problems.

Furthermore, it is easy to proof that the conditions 1-3 are sufficient for $f(x,y)$ to be a function of bounded variations.

In one-dimensional case, if we hold fixed one of the variables (x or y), then $f(x)$, $x \in [a,b]$ is a function of bounded variation. Variation is defined by:

$$\overset{b}{\underset{a}{V}}(f) = \sup_{x_1,\ldots,x_n} \sum_{k=2}^{n} |f(x_k) - f(x_{k-1})|, \tag{1.9}$$

where $a < x_1,\ldots,< x_n < b$ is an arbitrary system of points on $[a,b]$. According to the Dirichlet condition, a function $f(x)$, $x \in [a,b]$ is a function of bounded variation on every final interval, on which it is bounded and has final number of maximum, minimum, and discontinuity points of the first type.

According to conditions 1-3 it is easy to prove that the function $f(x)$ appears to be a function of bounded variations also for two-dimensional case. It is correct for the most common definitions of many-dimensional variations. Note, that there are many determinations of variation of a function of several variables. The majority of them rise to the determination of one functional, which boundedness guarantees the existence of certain characteristics of the function. In any case the list of these characteristics is too poor in comparison with one-dimensional case. It holds true for the Arzela, Vitali, Tonelly, and other known variations [2]. Essentially different approach was proposed by Kronrod, published in [2], where it was shown that a function of two variables must be characterized not by one, but by two independent functionals. Those functionals were determined as following:

$$w_1(f) = \int_{-\infty}^{\infty} v_0(e_t)\,dt, \qquad w_2(f) = \int_{-\infty}^{\infty} v_1(e_t)\,dt \tag{1.10}$$

where a set e_t is a t-level of the function $f(x,y)$; $v_0(e_t)$ is a number of components of the set e_t; $v_1(e_t)$ is a length of the set e_t [2]. For a continuous differentiable function:

$$w_2(f) = \iint_D |grad(f(x,y))|\,dxdy.$$

It is known, that numerous attempts to create image model basing on determination of only one functional cannot give us a complete description of image characteristics. Moreover, first component w_1 of two-dimensional variation in contrast to other metrical characteristics reflects topological properties of an image.

The class of functions of bounded variation is very extensive. In spite of this, the functions of such class possess many good properties. They have a total differential almost everywhere, their Fourier series converge to them almost everywhere, etc. [2]. In the physical applications, the condition of the boundedness of variations on every bounded domain means that $f(x,y)$ is bounded, and high frequency Fourier components cannot affect to the function intensity. It is interesting to note that mathematical description of wave diffraction leads to the intensity distributions, which can be used as an illustration of the functions of unbounded variations [7].

1.2.2 Metric characteristics on an image set

In the previous section we defined some limitations to the images. To operate with images as with mathematical objects, we have to introduce also the quantitative characteristics of images. At the beginning we have to formulate what is wanted to measure, and how to perform it. The first is how to compare images, i.e. how to metrize the image set. According to the aim of investigation, it is possible to construct different definitions of a distance between two images. However, an image in our interpretation is an information for visual perception. So in ideal case, a distance between two images must be measured correspondingly to some metric, being adequate to a human vision.

In spite of intensive study of human visual system, any satisfactory metric coordinated with subjective estimation it is not suggested. Moreover, the existing analogs of visual metrics [1] are too

complicated, or there are not metrics in the mathematical sense. Because of that, known visual metrics can not be used in optimization methods. Thus, the integral metric is most often used:

$$\rho(f,g)=[\int_D |f(z)-g(z)|^p dz]^{1/p}. \quad (1.11)$$

It is most simple to use (1.11) under $p=2$, because only in this case we can construct linear optimization algorithms.

In image processing, especially in decision of ill-posed problems, the norms of functions are used as the Tikhonov stabilization functional [4, 5, 6]. Different norms of functions can be used as a measure of image smoothness. As it follows from the definition, a norm measures the distance to zero, that is a distance to the most smoothing function. The quadratic forms are usually used to avoid the nonlinearity. For example, the set of images is supposed to be the Sobolev space $W_p^l(D)$ with respect to the norm:

$$\|f\|_W^2 = \iint_D [af^2 + b_1(\frac{\partial f}{\partial x})^2 + b_2(\frac{\partial f}{\partial y})^2 + c_1(\frac{\partial^2 f}{\partial x^2})^2 + c_2(\frac{\partial^2 f}{\partial x \partial y})^2 + c_3(\frac{\partial^2 f}{\partial y^2})^2]dxdy. \quad (1.12)$$

In discrete approximation, the usage of a Gaussian image model leads to the minimization of the following quadratic forms:

$$\|f\|_G^2 = (f-\mu, C^{-1}(f-\mu)) \quad (1.13)$$

where C is a covariance operator of $f(x,y)$, μ is a mean value. The norm (1.13) corresponds to a Sobolev norm (1.12) under certain conditions [5]. It is necessary to note that in some cases the usage of quadratic forms in the extreme problems can lead to the undesirable results.

Besides norms, all measures of smoothness, which are defined on the functional set, can be used for the decision of decomposition problem. The most popular is the probability measure $p\{z\}$, which is introduced on a set of discrete images [1]. Considering an image two-dimensional geometric surface $z=f(x,y)$, one can use any of the geometric measures or other metric characteristics, for example, entropy and variations. In particular, the Hausdorff measures were efficiently adapted to the fractal theory. It is known that the variations of a set give more adequate description of its geometrical structure in comparison with measures and entropy [2]. For the estimation of the geometric "complexity" of an image the variations should be used. Correspondingly to real images the complexity reflects "unsmoothness" of an image. So, increasing image's complexity we reduce its smoothness, and vice versa. Very often we use such a measure as a power of a finite set, for example when an image's histogram is estimated. A truth value is applied as a measure in some special cases, when we must say "yes" or " no" using video data.

The main question is the following: which metric characteristic of the whole variety existing in modern mathematics to choose. In other words, there is a need to estimate the quality of metric characteristics, which is defined as the usefulness of the measures for a given purpose and a given user. Unfortunately, except some general reasoning we can propose only the numerical experiments to answer this question. In the next section we describe the scheme that can be used in numerical experiments to compare the effectiveness of the usage of different metric characteristics .

2. IMAGE DECOMPOSITION AND ENHANCEMENT

2.1. IMAGE DECOMPOSITION

The image smoothing operation may be treated as some transformation, which lefts on an image only lengthy details, and avoid small details, noise, texture, etc. To understand this process better, we interpret image as a combination of statistically and semantically independent components, each of them containing details of different information classes. Statistic distinctions of them allow to find an algorithm for extracting from an image only the component of interest. Such splitting an image to the components we call *image decomposition*.

We will find rank algorithm to decompose an image to two components: smoothed one, contained lengthy details with sharp edges, and texture one, contained fine details, texture, and noise. These components may be used both separately for the consequent analysis (for example: image smoothing, object extraction, texture analysis, etc.), and together for the image enhancement.

According to (1.1), the decision of image decomposition task consists of finding the value of smoothed component s_{mn} for each image pixel. We will find decomposition algorithm in the class, based on analysis of image brightness distribution for local fragment, rounding current processed elements. According to piecewise image model [12], the probability density of brightness $\mathbf{P}\{x\}$ for the fragment W_{mn}, containing R domains $U^1,...,U^R$, has R local maxima (see Fig.1). Their positions coincide with the values of mean brightness' of each domain: $S^1,...,S^R$. We assume, that each image pixel belongs to some domain r. Also s^r_{mn} is the mean value for the part of domain r, containing this pixel, and placed inside the fragment W_{mn}. So, in a distribution $\mathbf{P}\{x\}$ we have to find the peak, corresponding to the domain rounding central pixel, and then determine its local maximum. The problem is that exact shape of distribution $\mathbf{P}\{x\}$ is unknown, and we only can arrange it by the sample probability – the local histogram $\mathbf{H}^W_{mn} = \{h(k)\}$, reflecting the fraction of pixels of value k, been contained inside the fragment W_{mn}.

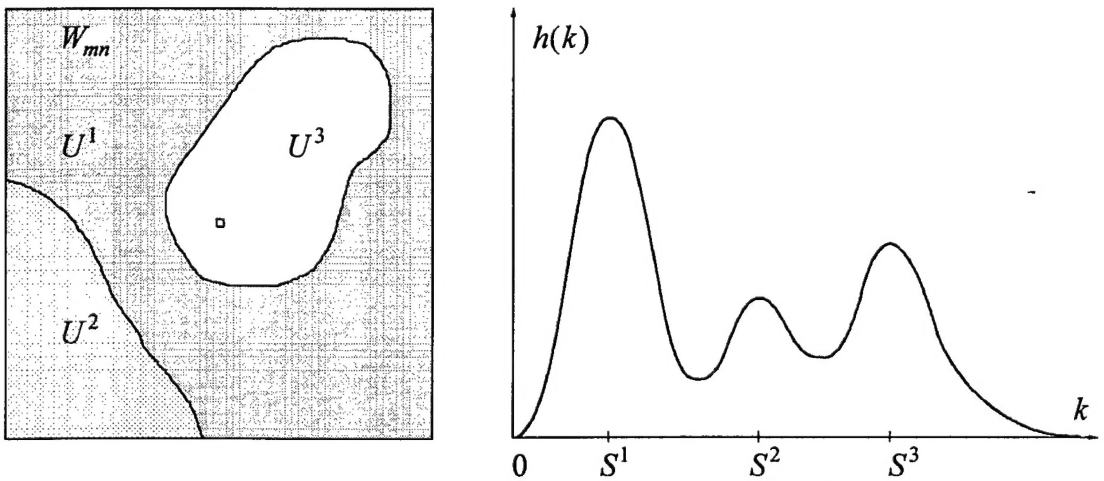


Fig.1. An image local fragment W_{mn} containing three domains (U^1, U^2, U^3), and its histogram $h(k)$.

We will find out image decomposition algorithm among rank ones, based on order statistics of the fragment. Most known and simple of them is local median (1.8), but performed analysis [12] demonstrates, that both local median and local average (1.7) give insufficient decision of the above task. Much better are algorithms, based on M-estimates and using order statistics [13]. This type of estimates is more adaptive to signal variability, and consists of the following. Let some weighted function $w_i(x)$, $(i = 1,...,n)$ is defined for the ordered set of elements $\{x\} = x_1 \le x_2 \le,...,\le x_n$ $(x_i \in W_{mn})$. M-estimate for this set will be:

$$M(W) = \sum_{i=1}^{n} x_i w_i(x_i) \Big/ \sum_{i=1}^{n} w_i(x_i).$$

If for any i we choose $w_i(x) = 1$ if $-\delta \le x \le \delta$, and $w_i(x) = 0$ for other x, then we obtain sigma filter Lee [14]. The value of central pixel of the fragment x_{mn} is chosen as a zero-point:

$$y_{mn} = \sum_{i=1}^{n} x_i w_i(x_i - x_{mn}) \Big/ \sum_{i=1}^{n} w_i(x_i - x_{mn}). \qquad (2.1)$$

This filter was proposed for image smoothing, and it was recommended to choose $\delta = 2\sigma$, where σ^2 is noise variance. In the survey [8] the sigma filter was nominated as one of the best edge-preserving noise cleaning filter for fragments of 5-7 elements' width. The weak place of sigma filter is the choice of zero point for weighted function $w_i(x)$. The value of central pixel of the fragment, as it was proposed in [14], is simple but too inaccurate estimate for noisy images. It is enough probable, that the difference $x_i - x_{mn}$ is so big, that desirable peak is out of the range $-\delta \le x_{mn} \le \delta$. So our estimate proved to be incorrect, will lose the accuracy, and filtered image will be blurred (this effect has more high influence near contour edges).

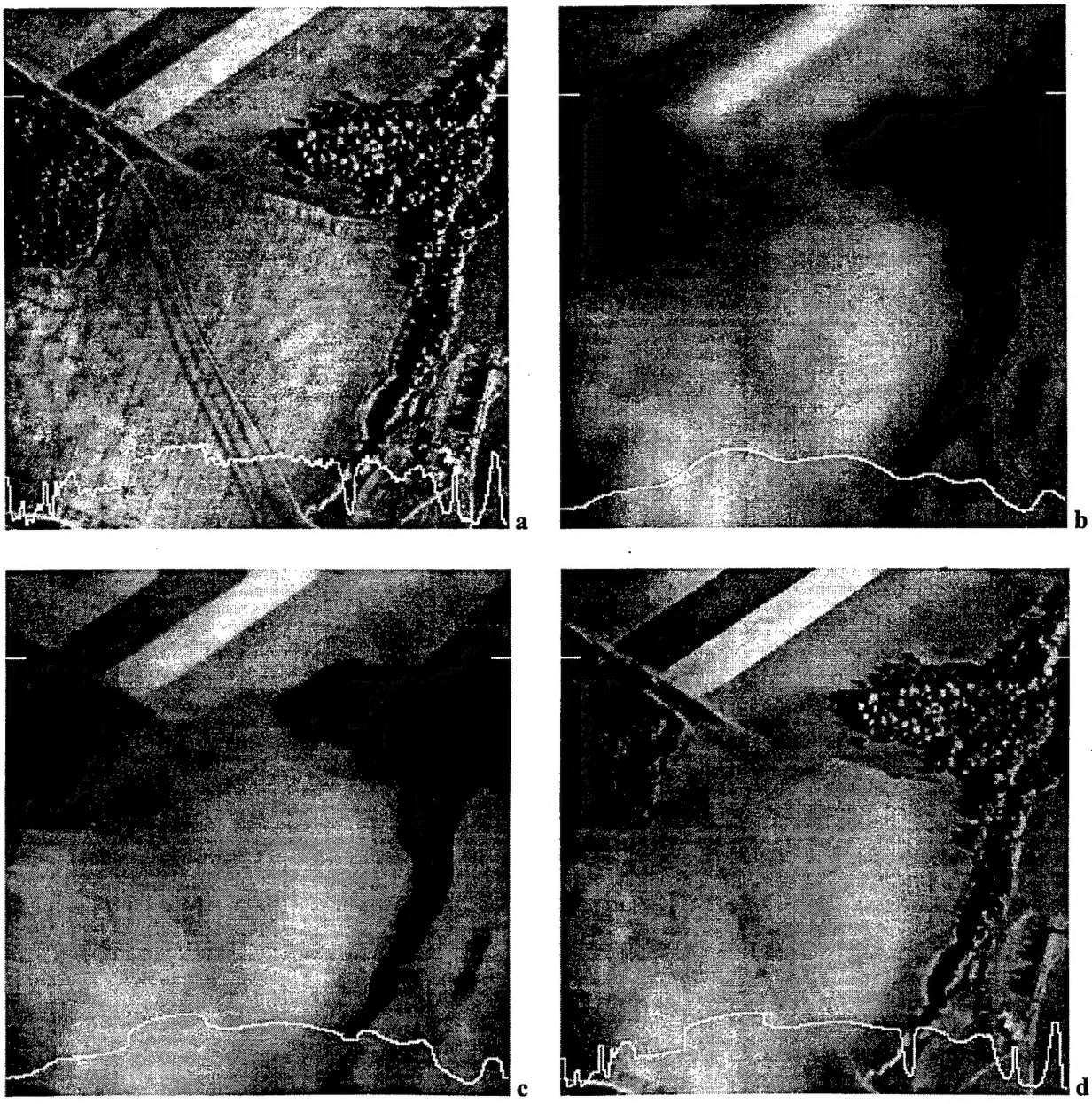


Fig.2. Comparison of image smoothing algorithms. Source airphoto image (a); its local average (b), local median (c), and smoothed component after the image decomposition (d).

Basing on the model (1.3) and on the representation of the fragment as a set of domains, we propose iterative algorithm, containing sigma filter as a subclass. The idea consists of iterative analysis of embedded fragments of different sizes. The smaller is the fragment, the less is the number of domains being covered, and hence the less is the influence of redundunt domains. Of course, the accuracy of the estimate will also be reduces correspondingly to smaller window size. Value, obtained for the smaller fragment, is used as a null-point for larger one. Thereby following iterative algorithm may be formulated.

On each iteration q, estimate y_{mn}^{q} is calculated over corresponding fragment W_{mn}^{q}. The value y_{mn}^{q-1}, obtained on the previous iteration over the fragment W_{mn}^{q-1}, is used as current zero-point of weighted function. For the first iteration the value of central pixel x_{mn} is used. The value, obtained on the last step Q, is assumed to be the resultant value of smoothed component at a point (m,n):

$$s_{mn} = y_{mn}^{Q}; \qquad y_{mn}^{q} = \sum_{x \in W_{mn}^{q}} x w(x - y_{mn}^{q-1}) \Big/ \sum_{x \in W_{mn}^{q}} w(x - y_{mn}^{q-1}). \tag{2.2}$$

Experiments demonstrate, that two iterations is usually enough: first one for the vicinity of 3-5 pixel width, and second one for the fragment of 15-25 pixel width. This algorithm has enough high accuracy in estimating of smoothed component on noisy images: processing of artificially composed signals (a set of flat objects with 5% gaussian noise) demonstrates mean square error near 0.3-0.5%. Visual estimates of images after the decomposition also confirm the efficiency of the algorithm. Qualitative estimate of images' decomposition with the helps of two-dimensional variation is discussed in the following chapter.

The example of smoothing some airphoto image with the helps of local average (1.7), local median (1.8), and proposed algorithm of image decomposition (2.2), is represented in Fig.2. One can see significant differences of contour sharpness in pictures (b),(c), and (d). These results demonstrate, that image decomposition algorithm (2.2) can successfully extract smoothed component from the image.

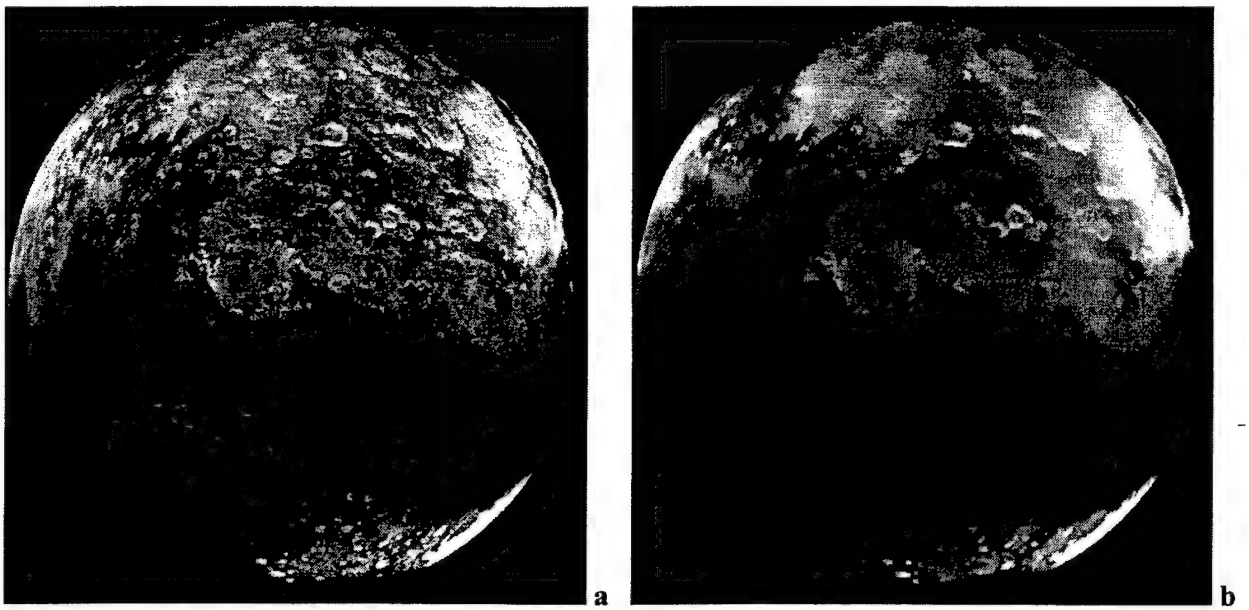


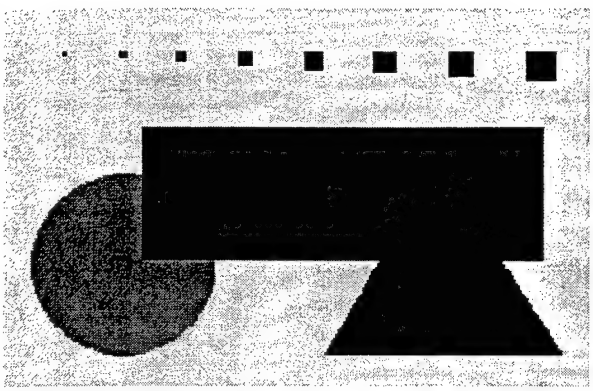
Fig.3. Example of image decomposition: a – source image (Mars surface); b – smoothed component after the decomposition.

2.2. IMAGE DECOMPOSITION AND TWO-DIMENSIONAL VARIATION

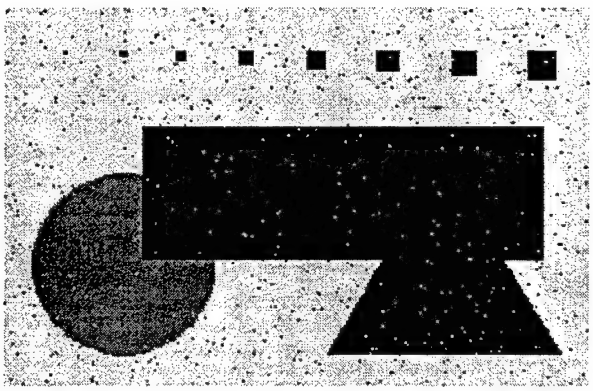
In all image smoothing algorithms the shape of operator and the power of smoothness is defined by some parameters. Most often the parameters are chosen empirically with visual estimate of a result, but in many cases they may be found as a solution of some optimization problem, so we need to measure the properties of images we are interested in. In our case such estimate is the smoothness of an image. Above we discussed formal possibilities of application of two-dimensional variation (1.10) for image estimation. Nevertheless one question remains unknown: does this estimate can reflect the complexity and smoothness of images better than other metrical characteristics, based on only one functional? To answer the question the set of experiments in smoothing some test images was performed.

For the numerical estimate and comparison of proposed image decomposition with traditional image smoothing, based on local average and local median, both artificial and real images were used. These images are presented on Fig.3. Four images were used for the testing: two real ones (Landscape and Portrait), and two artificially generated in following manner. A priory smooth artificial image (piecewise constant signal – a set of domains of different sizes and brightness) was dusted initially by 2% gaussian noise (Test Image 1). Second image was dusted by a mixture of 2% gaussian and 2% uniformly distributed pepper-and-salt noise to make a noise with "heavy tails" (Test Image 2). These images were smoothed by image decomposition algorithm (8) with different outer windows and thresholding parameters. Well known moving average and moving median algorithms were used for the comparison. Two-dimensional variations (1.10), mean square deviations from the original signal, entropy, and differential entropy were measured. These data are reduced into the Table 1 and are drawn as a set of plots in Fig.3 and 4.

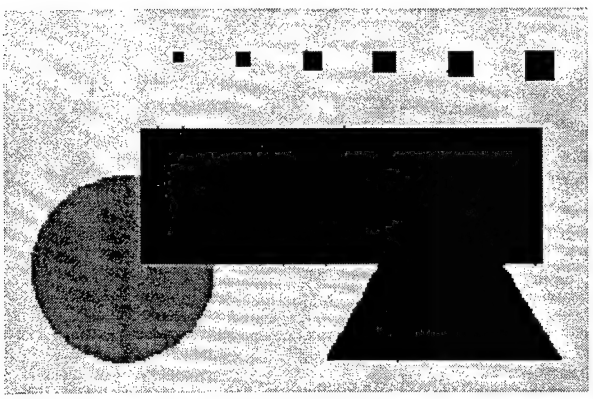
For the beginning it is interesting to compare image decomposition with local average and local median. Mean square error on Fig.3.b and 3.e shows, that image decomposition restores original image enough precisely, whereas local average and median significantly distort an image (Fig.3.b). The smoothness of images after the transformations may be reflected by difference entropy (Fig.3.c); one can see, that decomposition produces more smoothed image. Along with that, the first component of two-dimensional variation (Fig.3.a) reflects, that both average and mean produces images without details (corresponding plots are tending to zero), whereas decomposition plot tends to some non-zero constant. Plots on Fig.3.d and 3.f demonstrate, that choosing of thresholding parameters helps to make different smoothed and detailed images.



a



b


c

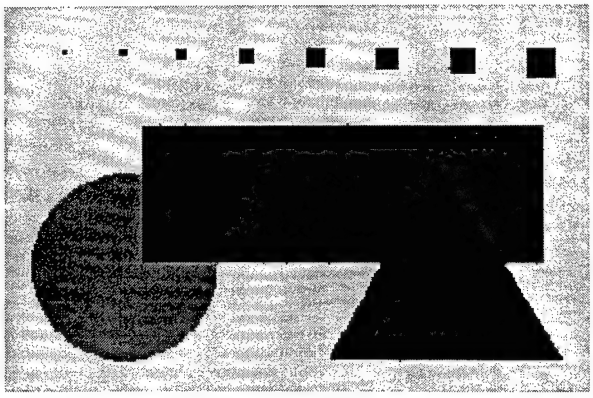

d

Fig.3. Test images: a – source image; b – test image 2 (corrupted by gaussian and uniformly distributed noises); c and d – smoothed component after the decomposition with different area thresholding.

Comparing the estimates of the first and the second components of two-dimensional variation one can see, that their behavior both on test and on real images looks similar. With increasing the size of outer smoothing window first component significantly reduces to some constant, while second component remains approximately the same (see Fig.4). The fact, that first component approximates to some constant (the plot has some "plateau") may help to choose the parameters of decomposition algorithm. It looks reasonable to recommend to take the parameters, corresponding the beginning of such plateau.

Perhaps it is early to draw any conclusions about global advantage of introduced two-dimensional variation, but in any case it looks perspective and demanding the continuation of researches.

Table 1. Values of two-dimensional variation: W_1 is the first, and W_2 is the second component.

	Source image		*Dusted image*			*Smoothed component (W1)*				*Texture component (W1)*			
Image	*W1*	*W2*	*W1*	*W2*	*MSE*	5	9	15	25	5	9	15	25
Test Image 1	6.74	695.2	463.7	2873	5	14.5	11.7	9.1	8.7	467	467	465	466
Test Image 2	6.74	695.2	795.3	4203	13.6	44.7	34.9	23.2	16.1	721	728	738	747
Landscape	–	–	133.6	1966	–	31.5	24.3	16.9	12.3	158	161	164	166
Portrait	–	–	252.9	3347	–	24.4	22.7	20.4	18.1	365	366	365	362

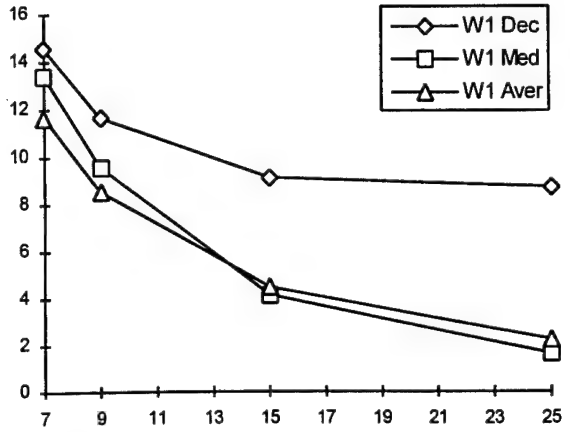
In the Tables 2 and 3 the following notation is used: *Fi* and *Fo* are sizes of inner and outer fragments; *Ti* and *To* are thresholding limits for inner and outer fragments; *C* is type of component: s - smoothed, t - texture (difference); W_1 and W_2 are the first and the second components of two-dimensional variations; *Mean* and *Var* are mean and variance; *Entr* and *DifEntr* - entropy and difference entropy.

Table 2. Image1 (Venus surface)

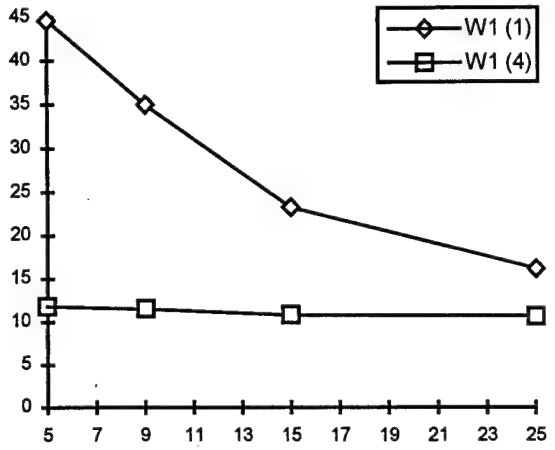
Fi	*Fo*	*Ti*	*To*	*C*	W_1	W_2	*Mean*	*Var*	*Entr*	*DifEntr*
Source image					133.56	1966.33	93.17	33.38	6.913	4.749
	7	Median		s	9.47	551.02	91.89	30.49	6.611	2.930
	7	Average		s	9.73	577.15	93.17	30.86	6.684	3.057
3	7	1	3	s	35.56	940.44	92.92	32.05	6.742	3.308
3	7	1	7	s	27.27	852.49	92.69	31.72	6.724	3.241
3	7	1	11	s	21.30	759.17	92.43	31.33	6.697	3.157
3	7	1	15	s	16.96	679.40	92.21	30.99	6.672	3.071
3	7	1	20	s	13.02	599.22	91.99	30.63	6.646	2.975
3	11	1	7	s	29.91	818.01	92.73	31.56	6.704	3.091
3	11	1	15	s	21.33	710.15	92.46	31.12	6.675	2.985
3	11	1	25	s	14.96	598.68	92.17	30.61	6.636	2.855
	7	Median		t	167.69	1832.59	129.29	11.05	4.779	4.641
	7	Average		t	162.47	1918.86	128.00	11.04	5.015	4.712
3	7	1	3	t	157.65	1429.38	128.26	6.03	4.477	4.372
3	7	1	7	t	160.24	1497.04	128.48	6.84	4.534	4.429
3	7	1	11	t	162.13	1584.20	128.74	7.92	4.602	4.491
3	7	1	15	t	164.31	1669.18	128.96	8.96	4.669	4.546
3	7	1	20	t	165.55	1767.51	129.96	10.20	4.759	4.608
3	11	1	7	t	147.43	1451.95	128.44	6.94	4.647	4.391
3	11	1	15	t	150.04	1540.09	128.71	8.10	4.716	4.456
3	11	1	25	t	151.33	1641.45	129.00	9.49	4.799	4.524

Table 3. Image2 (Airphoto)

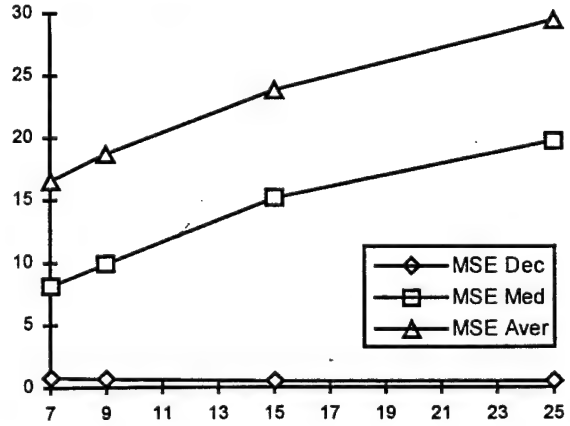
Fi	*Fo*	*Ti*	*To*	*C*	W_1	W_2	*Mean*	*Var*	*Entr*	*DifEntr*
Source image					252.94	3347.47	73.70	42.91	6.422	5.041
	9	Median		s	10.25	1145.58	73.22	40.24	6.283	3.175
	9	Average		s	9.49	1080.68	73.70	38.43	7.085	3.445
3	9	1	5	s	24.38	1658.67	73.59	41.83	7.172	3.647
3	9	1	9	s	22.67	1639.41	73.57	41.79	7.171	3.634
3	9	1	15	s	20.43	1579.23	73.52	41.67	7.167	3.604
3	9	1	25	s	18.09	1432.29	73.42	41.34	7.155	3.513
3	9	4	5	s	21.46	1590.45	73.45	41.71	7.168	3.610
3	9	4	9	s	20.45	1581.02	73.45	41.69	7.168	3.602
3	9	4	15	s	18.46	1537.60	73.41	41.60	7.164	3.579
3	9	4	25	s	16.70	1408.51	73.34	41.30	7.154	3.497
	9	Median		t	348.43	3000.87	128.48	12.06	5.087	4.921
	9	Average		t	320.99	3170.16	128.00	13.71	5.490	5.003
3	9	1	5	t	364.88	2416.84	128.11	6.09	4.605	4.676
3	9	1	9	t	366.16	2434.32	128.13	6.21	4.619	4.686
3	9	1	15	t	365.85	2487.20	128.18	6.67	4.667	4.714
3	9	1	25	t	362.02	2639.31	128.28	8.22	4.808	4.792
3	9	4	5	t	386.36	2557.07	128.25	6.65	4.676	4.751
3	9	4	9	t	386.09	2562.30	128.25	6.72	4.681	4.754
3	9	4	15	t	381.78	2588.33	128.29	7.03	4.713	4.767
3	9	4	25	t	373.29	2706.31	128.36	8.41	4.835	4.824



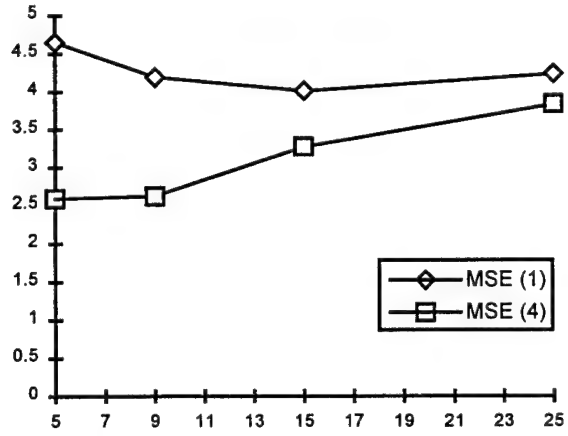
a. First component of variation (various smoothings)



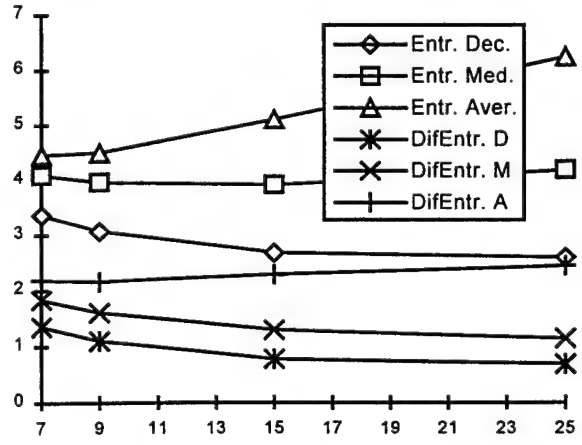
d. First component of variation (various thresholdings)



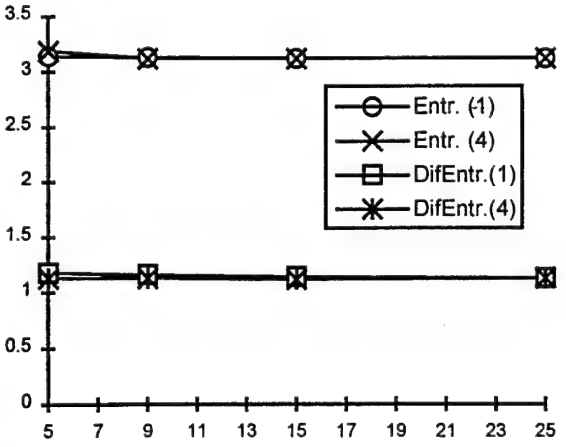
b. Mean square error (various smoothings)



e. Mean square error (various thresholdings)

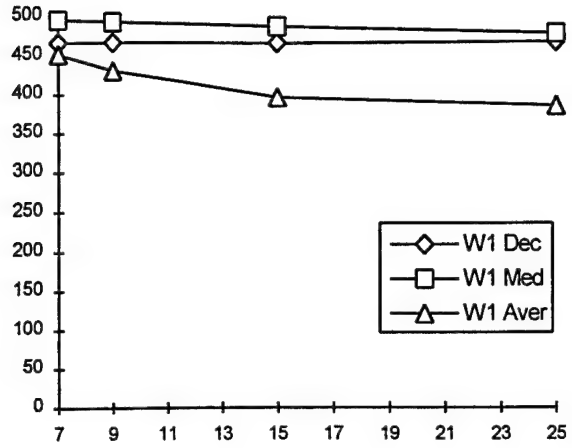


c. Entropy and differential entropy (various smoothings)

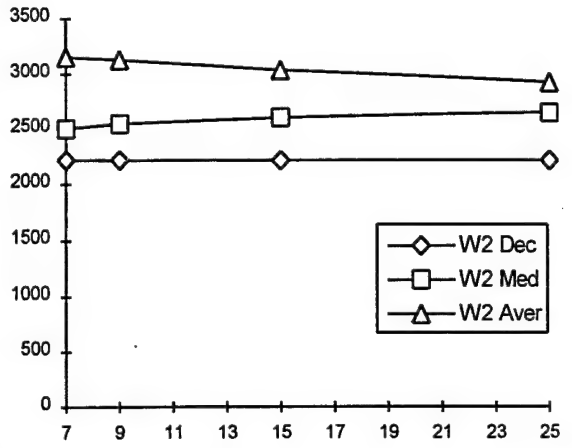


f. Entropy and differential entropy (various thresholdings)

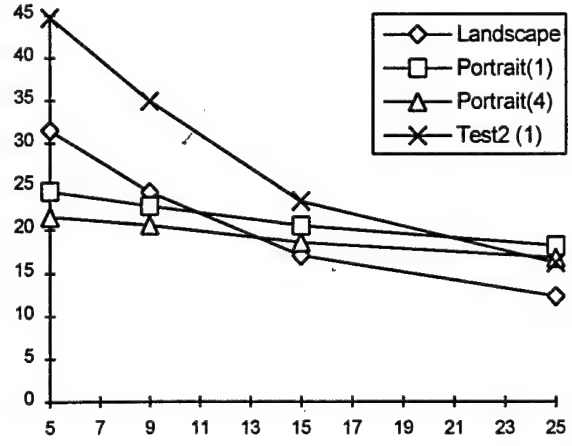
Fig.3. Dependencies of various estimates on the size of smoothing window (on X-axis): **a** and **d** - the first component of two-dimensional variations (W1), **b** and **e** - the mean square error (MSE), **c** and **f** - the entropy (Entr) and the differential entropy (DifEntr); **a**, **b**, and **c** - various smoothing windows (Test image 1): decomposition (Dec), median (Med) and average (Aver); **d**, **e**, and **f** - image decomposition under various thresholding (1 or 4) on outer window (Test image 2).



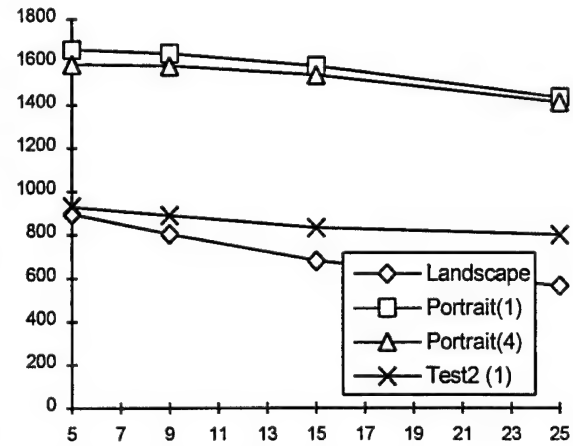
a. First variation
(Test Image 1, texture component)



b. Second variation
(Test Image 1, texture component)



c. First variation on real images
(smoothed component)



d. Second variation on real images
(smoothed component)

Fig.4. The behavior of two-dimensional variation on texture component and on real images after the decomposition; plots **a** and **b** shows the first and the second component of Test Image 1; plots **c** and **d** shows the first and the second component of real images in comparison with Test Image 1.

2.3. IMAGE ENHANCEMENT ON THE BASIS OF IMAGE DECOMPOSITION

Image enhancement is intended for rising of visual image quality before the following analysis. Rozenfeld [15] interpret image quality as spatial resolution and sharpness. Avoiding problem of increasing of resolution, and not taking into account independent problem of color reproduction, one may say, that visual image quality is dictated by the contrast and the sharpness of its details. Traditionally the problem of image enhancement may be subdivided into two parts: the *gray scale correction*, globally affecting to the entire image, and the *image sharpening*, being attained by local contrasting [1]. In the present paper we consider the second one, and treat it as magnification of local contrasts.

There are known many methods for image enhancement. Some reviews may be found in [1,8,9]. Also there is demonstrated, that better results are given by methods, adaptable to statistical parameters for local window. Most of them use local mean and variance, or similar estimates, like median and

interquartile distance [16]. The overwhelming majority of used methods may be generalized by some common filtering formula, closed to well-known unsharped masking. Detailed discussion will be done below, here we will only notify, that most important element of these methods is obtaining of well-smoothed image.

Image enhancement is based on human perception and visual estimation of an image, and is closely connected with extracting of video information from an image for the following analysis and interpretation. Because of the absence of formal image quality criterion, there is no global image enhancement theory, and it is usually treated as a raising of visibility of image details. This is attained by magnification of contrasts (both global and local) on an image. We consider here only the task of magnification of local contrasts.

Many methods for image enhancement are published. The comparison of used methods [1,8,12] demonstrates, that better results are given by methods, adaptable to statistical parameters for local window. Methods, analogous to well-known unsharp masking are most frequently used. In a simple form they may be described by the formula:

$$y_{mn} = K(x_{mn} - \tilde{s}_{mn}) + b\tilde{s}_{mn} + c,$$

where $\tilde{s}_{mn}$ is or local average or local median; K is coefficient of magnification of local contrasts, b and c are parameters of transformation (usually constants). By choosing K and b we may significantly change the properties of the filter. For example, if $\tilde{s}_{mn}$ is local average, $K = 0$, and $b = 1$, we obtain low-pass filter (image smoothing); if $K = 1$, and $b = 0$, we obtain high-pass filter; if $K > 1$, and $b = 1$, – local contrasts' magnification.

In a widely referenced paper [17], Wallis proposed a statistical operator for generalizes unsharp masking and statistical differencing [15] by forcing a desired mean and standard deviation:

$$y_{mn} = \frac{\alpha\sigma_d}{\alpha\sigma_{mn} + \sigma_d}\{x_{mn} - \bar{x}_{mn}\} + \beta\bar{x}_{mn} + (1-\beta)m_d. \qquad (2.3)$$

Here $\bar{x}_{mn}$ and σ_{mn} are local mean and variance of the source image; m_d and σ_d are desired global mean and variance of processed image; α and β are local contrast and brightness forcing factors. Analogous filtering algorithm was proposed in [16], where local median and interquartile distance were used instead of local mean and variance.

Some analysis and classification of different enhancement algorithms were done in [12], where they were reduced to closed form, generalized in the following manner:

$$y_{mn} = f(x_{mn} - \tilde{s}_{mn}, \sigma_{mn}) + b\tilde{s}_{mn} + c. \qquad (2.4)$$

Parameters, being presented in the formula, are related to following image properties: $\tilde{s}_{mn}$ - estimate of image brightness inside local window, rounding processed element (local mean, median, or similar); σ_{mn} - estimate of image variability inside the window; $f(u,v)$ - some monotone function of u, depending on parameter v.

Formula (2.4) describes enough wide class of transformations, definable by the choice of estimate $\tilde{s}_{mn}$ and function $f(u,v)$. Of course, estimate $\tilde{s}_{mn}$ may be not only local average or median. Looking to the formula it is evident, that $\tilde{s}_{mn}$ is very similar in sense to s^r_{mn} in (1.3). Such approach to image enhancement means nothing more than separating of source signal to two quite different components, and following transformation over them. So, we may substitute smoothed s_{mn} and detailed t_{mn} components, obtained after decomposition, into (2.4) instead of $\tilde{s}_{mn}$ and $x_{mn} - \tilde{s}_{mn}$.

Image enhancement basing on the decomposition (2.2) mainly demonstrates good experimental results. Nevertheless false contours sometimes rise on an image near not very sharp edges. The explanation of this artifact was done in [9] – the effect appears due the inaccuracy of piecewise constant model (1.3) for the non-sharp contour edges. Decomposition algorithm smoothes an image so, that it becomes to be even more sharp on contour edges, then source one. Hence, on such places the difference

$x_{mn} - s_{mn}$ will get opposite sign in comparison with $x_{mn} - \bar{x}_{mn}$ (element value minus local average or local median), and magnification of this difference induces the appearance of false contours. To avoid this lack it is enough to perform small linear averaging of smoothed component s_{mn}. Usually averaging inside the vicinity 3x3 is enough. Experiments on such enhancing some airphoto images basing on image decomposition demonstrate high quality of obtained results.

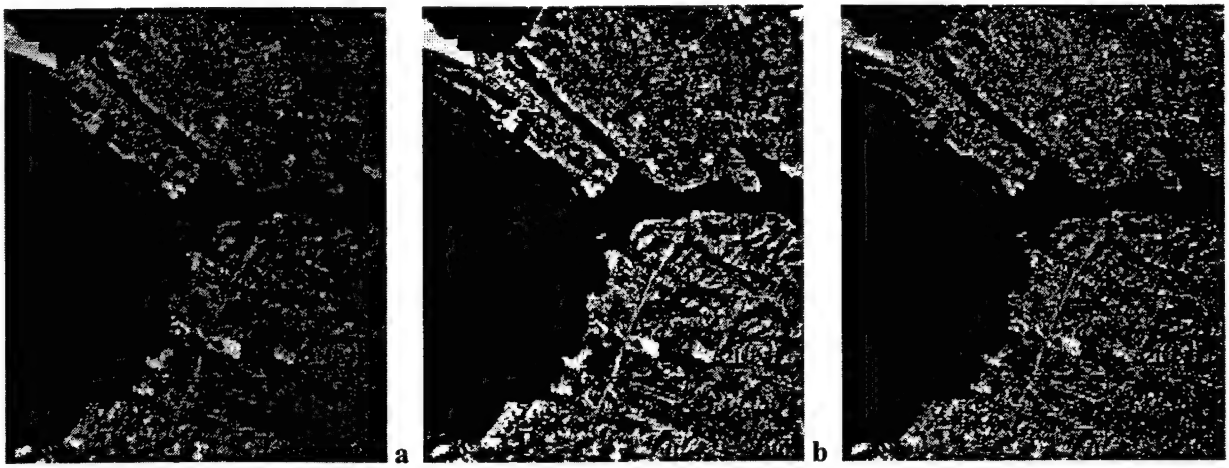


Fig.4. Enhancement of Landsat image (a) using local average (b), and smoothed component after image decomposition (c).

An example of enhancing some space image is demonstrated on Fig.4. One can see, that after image enhancement with local average (image Fig.4,b) small details and contours of expanded areas are overcontrasted, whereas after preliminary image decomposition (Fig.4,c) these lacks are avoided, and objects' brightness remain unchanged.

2.4. AUTOMATIC GRAYSCALE IMAGE CORRECTION

Automatic gray scale correction of captured video data (both still and moving images) is one of the least researched questions in the image processing area, in spite of this question is touched almost in every book, concerned with image processing. Classically it is related to the image enhancement [1,18,19], and frequently is classified as histogram modification techniques. Traditionally used algorithms, based on analysis of image histogram [20-23], are not able to decide the problem properly. The investigating difficulties are associated with the absence of formal quantitative estimate of image quality – till now the most often used criteria remain the human visual perception and experience. As a hence, is still unsolved the problem of finding out some measurable properties of real images, which might be the basis for automatic building of *gray scale correction function* (sometimes been identified also as *gamma-correction function*).

Below we will try to discern some common properties of real images, that could help us to evaluate the gray scale image distortion, and finally to construct the appropriate correction function to enhance an image. Such a method might be sufficiently used for automatic image processing procedures, like enhancing of medical images, reproducing of pictures in publishing industry, correcting of remote sensing images, preprocessing of captured data in computer vision area, and for many other's applications. The question of complexity of analysis procedure becomes to be important when an algorithm is realized in real-time (for example in video input devices, like video cameras).

24.1. Statement of a task

The goal is to propose some methods for automatic image enhancement by use of gray scale correction. We guess, that the only accessible initial information is already distorted captured image $x_{mn} \in X$, and formal representation of such gray scale image distortion is the following:

$$x_{mn} = f(y_{mn}). \tag{2.5}$$

Here y_{mn} and x_{mn} are pixels' values correspondingly of non-distorted original and of registered images, and $f(z)$ is the distortion function. Assuming that $f(z)$ is monotone and univalent, there exists an inverse function $F(x)$, which may be used to reconstruct source image. Its formal representation is also extremely simple:

$$y_{mn} = F(x_{mn}), \tag{2.6}$$

where $F(x)$ is gray scale correction function, destined to compensate the characteristics of video capture and storage devices, and to enhance the received video data. Nevertheless the simplicity of the formula not reflects the simplicity of the task.

The central problem here is the question of how to find out the function $F(x)$. In most cases there is no any usable information other then already distorted video data, and the direct image distortion function usually also remains unknown. Thus, the only way to find out the function $F(x)$ becomes the analysis of the source image. So, the main question is the criterion of estimating the image gray scale distortion.

Almost all known methods for gray scale image correction use only a distribution of pixel values (image histogram), and not take into account statistical correlation between neighbor pixels, though it is obvious, that gray scale distortions will change them too. Below we will demonstrate the possibility to construct gray scale correction algorithm on a basis of these statistical properties. For more deeply discussion of this question we must base on some image model, been adequate for our task.

2.4.2. Discussing image model

Real images are characterized by presence of a set of different size areas (details of a scene) with approximately constant or weakly varying brightness inside, been separated by contour borders. Very often an image is treated as some two-dimensional surface, been traced (with predetermined accuracy) across the values of image pixels in a coordinate plane (m,n) [1,24,25]. Lengthy image details are represented here as smoothed and approximately horizontal areas, and contours – as narrow strips with steep slope angle of the surface.

A suitable image model was proposed and researched above. Correspondingly to the model (1.4), for every image pixel x_{mn} there may be selected some closed vicinity V_{mn} (possibly of changeable shape) containing the set of neighbor pixels $x^r_{mn} \in V_{mn}$; $r = 1,...R$, belonging to the same image object or contour, as x_{mn} pixel. An appropriate plane may be drawn across the elements of the vicinity, which will be sloped relatively to the horizontal plane for the two-sided angle ψ_{mn}. Thus to each of the image points there may be assigned some vector $\mathbf{g}_{mn}$ with its amplitude $g_{mn} = tg\psi_{mn}$ and twist angle φ_{mn}. Correspondingly to formula (1.4), μ_{mn} is the value of drawn plane in the central point of vicinity (m,n), in other words the average value; ρ is Euclidean distance between the central point and the point at position r; and γ^r_{mn} is some random value, been defined by the granularity of registering materials and by the noise of input device. The value g_{mn} defines the *local contrast* inside the vicinity.

Calculate the dependence of the value g on the signal brightness by averaging it for a set of vicinities v_k, having the same average values $\mu_{mn} = k$, and let $N(v_k)$ is the number of such vicinities on an image:

$$c(k) = \frac{1}{N(v_k)} \sum_{x_{mn} \in v_k} g_{mn} . \qquad (2.7)$$

This function $c(k)$ we shall call the *Local Contrast Function*. In [24], where it was demonstrated, that local contrast function $c(k)$ for the whole image may be well approximated as a sum of independent distributions for two different sets of image pixels: contour element pixels, and internal flat areas' pixels. Each of them may be well represented by appropriate normal distribution, where the variance for contour pixels is 5-20 times more than the variance for flat areas.

2.4.3. Hypothesis about local contrast function

Substitute the formula (1.4) to (2.5). Average value μ_{mn} will change to $f(\mu_{mn})$, and component $\rho^r g^r_{mn}$ to $(f'(\mu_{mn}) + o(f'(\mu_{mn})))\rho^r g^r_{mn}$, where $f'(x)$ is the derivative of $f(x)$. The component γ^r_{mn} will not change, because we suppose that it was added after the gray scale distortion. Assuming that $o(f'(\mu_{mn}))$ is negligible small, we obtain:

$$x^r_{mn} = f(\mu_{mn}) + f'(\mu_{mn})\rho^r g^r_{mn} + \gamma^r_{mn} . \qquad (2.8)$$

For us it is important, that the tangent of slope angle of the plane, been drawn through the elements of vicinity, will change to the following value:

$$g = f'(\mu_0) g_0 . \qquad (2.9)$$

Substituting (2.9) to (2.7) for the image with initial local contrast function $c_0(k)$ we obtain:

$$c(f(k)) = f'(k) c_0(k) . \qquad (2.10)$$

Since we suggest that for $f(k)$ there is an inverse function $F(k)$, we may write the inverse equation:

$$c_0(F(k)) = F'(k) c(k) . \qquad (2.11)$$

From this equation it is evident, that for determining the gray scale correction function $F(k)$ we need to estimate the present local contrasting function $c(k)$ of distorted image, moreover we must to make a conclusion about the shape of so-called *ideal* local contrasting function $c_0(k)$.

What will happen with local contrast function $c_0(k)$ if we will take into account only some subset of image pixels? According to the image model (1.4), statistical properties of the value g_{mn} will significantly change depending on which set of image pixels the vicinity V_{mn} belongs to: to the set of pixels from flat regions, or to the set of contour pixels. Correspondingly to that, the shape of function $c_0(k)$ will also depend upon the chosen set of pixels: internal or contour ones.

The set of internal pixels of flat regions possesses the next two deficiencies: i) statistical properties of textures of different regions may significantly vary from one another, hence g_{mn} and $c(k)$ will reflect more these properties, but not actual gray scale distortion; ii) due to comparatively small variation of texture component there is enough high probability that the distribution of $\mathrm{P}_\mu(k)$ for internal pixels may contain some gaps, so $c(k)$ will be insignificant for these values of argument.

The set of contour pixels looks preferable for using. Values of g_{mn} for contour vicinities depend on the module of brightness differences $|s^i - s^j|$ for closed regions and on the width of contour overfall. Of course, absolute values of g_{mn} also depend on the sharpness of the image and on its discretisation, but these factors affect uniformly and linearly for all image pixels, and thus will be compensated. Under the condition that contour set of pixels is enough representative, we may suggest: i) the probability $\mathrm{P}\{\mu = k\}$ $(k \in [s^i, s^j])$ for the set of pixels, belonging to the contour between any two regions U^i and U^j, is distributed uniformly in the interval $[s^i, s^j]$; ii) the sharpness of contours not depends on the

values s^i and s^j, and may be assumed to be identical on the range of brightness; iii) the set of intervals $\left[s^i, s^j\right]$ covers the whole range of image brightness $\left[k_{\min}, k_{\max}\right]$.

From these suggestions we may conclude, that for non-distorted high quality image the values of local contrast function $c(k)$, been measured for the contour set of pixels, will be uniform for the whole range of image brightness $\left[k_{\min} \le k \le k_{\max}\right]$. Thus we may accept the hypotheses of uniform distribution of local contrast function for contour image pixel: $c(k) = C_0$.

Some experiments, been performed over the set of high visual quality images, demonstrate, that local contrast function $c(k)$, been measured for the set of contour pixels, is enough closed to uniform. On the other hand the same function, been measured for all image points, hardly correlate with image histogram – the peaks of histogram correspond to the depressions of $c(k)$. It is induced by influence of image texture, contained inside flat regions.

2.4.4. Local contrasts alignment algorithm

Relying on the hypotheses done above, that $c(k) = C_0$, and on equation (2.11), we may write: $F'(k) = C_0 / c(k)$. Calculating the sum from 0 to k we find:

$$F(k) = C_0 \sum_{i=0}^{k} \frac{1}{c(i)}.$$

Introducing the normalization for the maximum brightness: $F(K-1) = K-1$, we obtain:

$$C_0 = (K-1) / \sum_{k=0}^{K-1} \frac{1}{c(k)}.$$

Now we may write down final equation for calculating the gray scale correction function $F(k)$ for alignment the local image contrasts:

$$F(k) = (K-1) \sum_{i=0}^{k} \frac{1}{c(i)} / \sum_{k=0}^{K-1} \frac{1}{c(k)}. \qquad (2.12)$$

The proposed algorithm was tested on several standard images. The results demonstrate, that visual quality of images, processed by the algorithm (2.12), is preferable then quality of the images, processed by other gray scale correction algorithms.

One serious deficiency of this algorithm consists of necessity of preliminary detecting of image contours. Really, the gray scale distortion will also affect to the results of contours' extraction, hence the decision of the task may not be correct. To overcome this problem we need or to develop some appropriate adaptive contour detection methods, or to find out some modification of the algorithm, which not demand similar operations.

2.4.5. Modification off the algorithm

If we can't separate source image to contour and flat components enough precisely, let's try to use union set of pixels without any preliminary splitting. In such a case we need to compensate the deficiencies, introduced by using complex statistics of the whole image. As it was already discussed above, if in (2.12) we take into account the whole set of image pixels, then the local contrast functions $c(k)$ have significant depressions for the argument values k, been corresponded to the brightness of lengthy image areas. These depressions inverse correlate to peaks of image histogram, and in such a case $N(v_k)$ in (4) may be revealed as the image histogram $H(k)$, e.g., a number of image pixels of brightness k.

There may be different decisions of how to compensate this influence, but one well known and enough simple variant is the introduction of some compensating component into the formula (4) by

adding relatively small value $\delta(N)$ to $H(k)$ in the denominator (N is the total number of image pixels):

$$c_H(k) = \frac{1}{H(k) + \delta(N)} \sum_{x_{mn} \in v_k} g_{mn} . \qquad (2.13)$$

Such an addition of $\delta(N)$ reduces $c(k)$ for small $H(k)$, and thus compensate the depreciation of $c(k)$ for large $H(k)$. The obtained $c_H(k)$ must be substituted to (2.12), as before.

2.4.6. Clipped histogram equalization

In some applications we may not use the local contrast alignment algorithm. There may be several reasons for that, one a situation may happen because of enough specific source data. For example, if captured image contains brightly illuminated object and hardly shadowed background without any details of intermediate brightness, there may appear well-expressed internal gaps in probability distribution of image brightness. Frequently the algorithm (2.13) is unable to remove such gaps. In other case the proposed algorithm may not satisfy us due to its relative complexity, for example if we need automatic gray scale correction, been realized inside video camera in a real time. For such a case the algorithm (2.13) looks to be too complicated. We must simplify the analysis procedure of the source image according to the limitations of weight, size, and price of such devices.

Thinking about this task in the context of image enhancement one may recognize, that more often an observer wants to compensate the deficiency of objects lighting, instead of to correct possible non-linearity of registering device. Moreover, when an image is used for a visual analysis, the observer also would like to get it with the best (usually with maximum possible) visual contrast of fine details, and frequently he/she would be ready to forgive some mistakes in differences of brightness between lengthy objects, if there are well distinguished. This facility may be used for working out comparatively simple but enough efficient algorithm. The goal of the algorithm is to magnify contrast in representative sections of gray scale range at the cost of compressing non-used ones, provided that the magnification may not exceed some boundary value.

We may formalize the task in the following manner. The probability distribution of image pixels' values is represented by it's (normalized) histogram $\mathrm{P}\{x = k\} = H(k)/N$, where $H(k)$ is the image histogram, and N is the total number of image pixels. Accordingly to the task, the whole dynamic range of image brightness $[0, K]$ may be separated to a set of intervals $[k_i, k_{i+1}]$, where $H(k)$ or has appreciable non-zero value (i.e., greater than some threshold of significance), or equals to zero (i.e., is less then the threshold). Permitting the existence of zero-length intervals we may accept, that $H(k)$ considered to be zero in every even interval $[k_{2i}, k_{2i+1}]$, and to be non-zero in every odd one $[k_{2i+1}, k_{2i+2}]$. Now it's obvious, that to solve the problem we may simply to discard all even intervals, then to join odd ones, and finally linearly stretch the derived union to the whole region of values $[0, K]$.

The formulated task may be decided with the help of different algorithms, and we would like to propose enough robust one. There is well-known algorithm – histogram equalization [23] – which has such ability to discard unused region of brightness. Its deficiency consists of hard nonuniformity of contrast transformation: the magnification for the brightness k is proportional to the probability of pixels of that value (in other words is proportional to the histogram value $H(k)$). To be applied to our task this algorithm must be modified in the following manner.

Let there are given two thresholds: T_0 as the threshold of significance and T_1 as the threshold of saturation $(T_0 \le T_1)$. If $\mathrm{P}\{k\} \ge T_0$, we will consider that the brightness k is significant and corresponding interval will be added to odd ones. Otherwise, if $\mathrm{P}\{k\} < T_0$, it will be added to zero-level (even) intervals. Additionally, if $\mathrm{P}\{k\} > T_1$, $H(k)$ will be clipped down to the level T_0. Thus we may write the next formula of transformation (clipping) the image histogram :

$$H_c(k) = \begin{cases} T_0 N, & if \quad H(k) < T_0 N; \\ H(k), & if \quad T_0 N \le H(k) \le T_1 N; \\ T_1 N, & if \quad H(k) > T_1 N. \end{cases} \tag{2.14}$$

Now it is possible to perform the standard histogram equalization algorithm [23] for the obtained function $H_c(k)$. The final equation for calculating *clipped histogram equalization* function $F(k)$ will be the following:

$$F(k) = (K-1)\sum_{i=0}^{k} H_c(k) / \sum_{k=0}^{K-1} H(k). \tag{2.15}$$

The advantage of this formula is that it demands very simple image analysis algorithm – the only calculating of the image histogram is needed, thus it may be successfully used in real-time applications.

Such a transformation was experimentally realized inside one digital video camera to enhance an image in a real time. The Histogram/Hough Transform Processor chipset L64250 (designated and manufactured by LSI Logic Corporation) was used to count the image histogram $H(k)$ during video frame. These data were transferred to a host processor to perform clipping and equalization correspondingly to formulae (2.14) and (2.15). The resulting transfer function $F(k)$ was downloaded back to the L64250 (as a look-up-table) to process video signal in a real time. The maximum value of delay between video data analysis and making the transformation active not exceeds two video frames.

Clipped histogram equalization was well tested both for static images and for real video. The visual quality of output images was acknowledged as well acceptable.

2.4.7. Results and discussion

Described above local contrast alignment algorithm has been sampled by transforming different images. There were tested three categories of images: i) originally distorted low-quality images, ii) standard high-quality images been artificially degraded by some gray-scale distortion, and iii) standard high-quality images in their initial view. All the images were enhanced by algorithm (2.13), and two estimates were defined for each the image: its general visual quality, and its relative quality in comparison with the source image (better or worse than the quality of original image).

The global estimate of applicability of the algorithm regards it as well sufficient for most number of images. Originally distorted images (i) were enhanced both by general and by relative estimates. Degraded images (ii) also where enhanced relatively to the source (degraded) copy, but in many cases their quality remains worse than the quality of initial (standard, not degraded) images. The standard images (iii) after the transformation make up three approximately equal groups: the group of images with almost identical gray scale transformation, the group of definitely changed images but with preserved visual quality, and the third one – the images of deteriorated quality. The explanation of this deterioration may be the following. High-quality images typically occupy the whole range of brightness, and initially they already possess high contrasts between lengthy objects. The algorithm (2.13) often tries to expand contrast of fine details in shadowed or bright areas, where original brightness may be saturated. However by expanding the contrast in some ranges of gray scale it will reduce the contrasts in other's ones. Thus the algorithm may induce the reduction of contrast between lengthy areas, that often is perceived as deterioration of general image quality.

We discussed above only the gray scale images, but proposed algorithms also may be applied for enhancing true-color images. To do that a color image must be represented in the form of brightness, hue, and saturation (if it is presented in usual form of red, green, and blue components, it must be correspondingly converted). Then the transformation (2.13) has to be invoked for the brightness components only, and the other's two should be remained as is. Then the image must be inverted back to the useful form. The performed experiments demonstrate the efficiency of this algorithm for enhancing visual quality of true-color images.

Local contrast alignment algorithm demonstrates its ability to enhance different kind of images, and it is preferable in comparison with other gray scale correction algorithms. In spite of it is unable to enhance any kind of source images, it may be recommended for most number of low-quality images. This algorithm may successfully supplement the local image contrasting algorithm (2.4).

The clipped histogram equalization algorithm (2.14) may be recommended for the applications, requiring real-time image processing. Such a transformation was successfully realized and tested in experimental video camera, and estimating of output image quality demonstrates the efficiency of the algorithm.

2.5. IMAGE RESTORATION

2.5.1. General principles of solving the restoration problem

The image restoration problem is usually formulated in the following way: to find (to estimate) an ideal non-degraded image z from the equation

$$Az + n(x,y) = u(x,y) + n(x,y) = \tilde{u}. \tag{2.16}$$

Here $u \in U$, $z \in Z$, A is a linear imaging operator, n(x,y) is the noise, and $u(x,y)$ is an output degraded image.

The most universal principles of solving that problem are formulated in the statistical estimation theory and in the theory of solving ill-posed problems [4]. Beside the general rules there exist numerous methods of restoration based on the usage of the specific features of the problem (a simple type of the distorting operator, existence of a known background, possibility of obtaining a large number of images of the same object, etc.). But regardless of what approach we use, the restoration problem can not be solved with the help of the so called empirical methods that are effectively applied to solve other problems of image processing (filtration, segmentation etc.). Restoration problem is a typical inverse problem of mathematical physics and as any other inverse problem can be solved only on the basis of exact mathematical methods.

2.5.2. General restoration formulas

What ever method we use to obtain the restored image, it must comply (in a certain way) with the main equation (2.16), i.e. it must provide the closeness of the left and the right sides in this equation. So the most general formulation of the restoration problem can de reduced to the functional minimization:

$$z^*(\varsigma,\eta) = \inf_{z \in Z} \rho_U(Az, \tilde{u}) \tag{2.17}$$

where ρ_U is a certain metric in U. To guarantee the uniqueness and the stability of the solution, it is necessary to formulate a priory information about an ideal image as a functional $\Omega(z)$ that possesses stabilizing properties. In this case

$$\begin{cases} z^* = \inf \rho_U(Az, \tilde{u}), \\ \Omega(z) \le C. \end{cases} \tag{2.18}$$

Under certain conditions the problem (2.18) can be reduced to the unconditional extreme problem, in particular to Tikhonov minimization [4]:

$$z^* = \inf_{z \in Z} \left\{ \rho_U(A, \tilde{u}) + \alpha\Omega(z) \right\}, \tag{2.19}$$

where a is the parameter of regularization. Usually it is assumed that the ideal image is a smooth function with respect to Sobolev space.

If U is defined as Euclidean space with respect to the norm (u,Bu), where B is a positive definite operator we obtain:

$$z^* = \inf_{z \in Z} \left\{ \|Az - \tilde{u}\|_B^2 + \alpha\Omega(z) \right\}. \tag{2.20}$$

It should be noted that the statistical methods used in image restoration result in extreme problems similar to (2.20). So using Bayes strategy or MAP test we obtain the optimal estimation in the form

$$z^* = \inf_{z \in Z} \{-\ln q(Az - \tilde{u}) - \ln p(z)\}, \tag{2.21}$$

where $p(z)$ and $q(x)$ are a priori probability densities of the ideal image and the additive noise $n(x,y)=(\tilde{A}z - \tilde{u})$. Using in (2.21) the most popular probabilistic models of the image and the noise specifically when $q(x)$ is Gaussian and $p(z)$ is Gibbs distribution, we obtain:

$$z^* = \inf_{z \in Z} \left\{\|Az - \tilde{u}\|_B^2 + U(z)\right\}, \tag{2.22}$$

where U is Gibbs potential and B^{-1} is covariance operator of the noise [5].

The main essential difference between the regularization method of image restoration (2.20) and the statistical method (2.22) is the existence of the regularization parameter in (2.20). It is necessary to point out that the opportunity of obtaining a family of solutions that depend on a parameter is very important. This allows us to control the visual quality of image restoration interactively regardless of a mathematical criterion of visual image quality.

Usually the linear algorithms are used in image restoration. These algorithms can be represented in the following form

$$z^* = \inf_{z \in Z} \left\{\|Az - \tilde{u}\|_B^2 + \alpha(z, Cz)\right\}, \tag{2.23}$$

here B and C are the positive definite operators.

We stress that the statistical methods, in particular Wiener filtration, lead to the minimization of the same quadratic functional (2.23) assuming that the image and the noise are Gaussian, and α=1. In this case B and C are inverse covariance operators of the noise and the image: $B=[Cov(n)]^{-1}$, $C=[Cov(z)]^{-1}$. From (2.23) it follows that linear algorithms differ from each other by choice of operators B and C. It can be shown, that as a rule, the result of linear restoration does not depend much on the operators B and C, i.e., the result does not depend much on the covariance characteristics of z and n. In any case the quality of the linear restoration always depends on the characteristics of degraded operator A and the magnitude of the noise $\|n\| = s$.

We propose to obtain an essentially new result using the non-linear method of restoration, when a priori information about the image is given not in the square functional form

$$z^* = \inf_{z \in Z} \left\{\|Az - \tilde{u}\|_B^2 + \alpha Var(z)\right\}. \tag{2.24}$$

Using the definition (1.10) it is possible to obtain the non-linear algorithm of image restoration:

$$\begin{cases} z^* = \inf \|Az - \tilde{u}\|^2 \\ w_1 \le C_1, \\ w_2 \le C_2, \end{cases} \tag{2.25}$$

or under certain conditions:

$$z^* = \inf_{z \in Z} \left\{\|Az - \tilde{u}\|^2 + \alpha_1 w_1 + \alpha_2 w_2\right\}. \tag{2.26}$$

So the decomposition of an image into two independent components can be effectively used in image restoration.

3. IMAGE COMPRESSION BASED ON THE WAVELET TRANSFORMATION WITH NONLINEAR DECOMPOSITION

3.1. Statistical and Visual redundancy.

The classical scheme of image compression (and being also applicable to image transmission and archiving) is shown on Fig.5. The source can be treated as an application or a memory where images are fetched from. Compressor reduces redundant information and sends it through the channel. Decompressor restores transmitted signals and transforms them into the form suitable for a viewer. The models of a source and a receiver determine the meaning of "redundancy". The first one describes so called statistical redundancy (S), and the second one – visual redundancy (V).

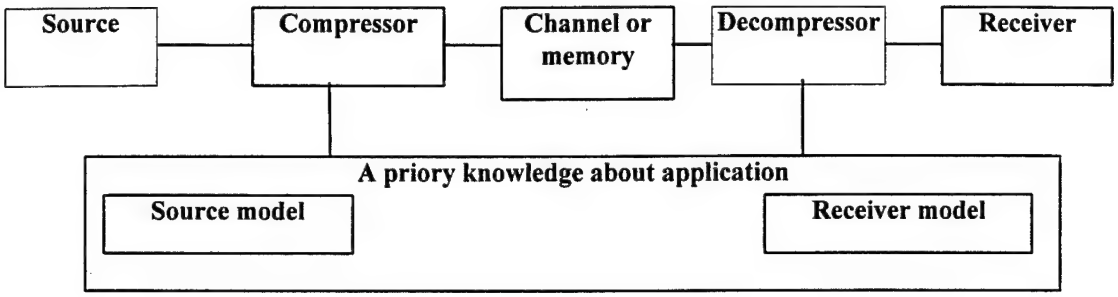


Fig. 5. Classical scheme of information transmission system.

3.1.1. S-redundancy. Source models.

The natural images are highly structured having components, which look like smoothed or flat areas surrounding by contours, textures and noise. Intuitively each of them separately can be described with reasonable precision by rather simple models. The images are complex compositions of those components. The task of a source model is to decompose the complex image on the independent components based on a priory information about images. The key to effective decomposition is that each one of the signal components may be represented efficiently by the model. So, the sufficient model should have corresponding number of appropriate components to represent as many of them as needed for an image receiver.

The degree of statistical independence achieved may be defined in terms of entropy. For that we have to compare the joint entropy among all the components with the sum of individual entropies of each the component (c_m is m-th component):

$$H(c_0, c_1, ..., c_M) \le \sum_m H(c_m). \qquad (3.1)$$

If there exist statistical dependencies among the c_m, then the joint entropy will be less than the sum of individual entropies, otherwise the two quantities will be equal. Assume that there is some way to verify that the total information in the image (joint entropy) is preserved, then we can reduce statistical dependencies by lowering the individual entropies $H(c_m)$ as much as possible. Statistical dependence can be even more complicated if between components Affine or other dependencies exist. Special case of such images, having high order S-redundancy, is self similar images, or their complex compositions, called fractals and multifractals. In this way due to the complex statistical dependencies among the components, the higher order redundancy is transformed into a simple first order redundancy. In such a case, the less entropy has each a component, the better is the performance of a system [35].

3.1.2 V-redundancy. Receiver models

The receiver model describes visual V-redundancy specifying the components being not distinguished from a background or other components by the receiver in a certain viewing conditions. Visual independence also means that varying one component we do not change visibility of others. That is why one of the task of a receiver model is to decompose an image to visually independent components.

The sensitivity of human vision model (HVS) according to the pixels within the component, usually is assumed to be equal or slowly varying.

It is known that the human vision has some thresholds, and the fidelity to image representation, demanded by the human eye, differs from pixel to pixel. The HVS has two types of thresholds: global one and local ones [32-34]. Global thresholds tune to the mean brightness of an image, and can slowly change from image to image. Local ones otherwise change quickly from pixel to pixel within the same image. This is called *masking effect* that can be temporal and spatial. Spatial masking increases thresholds for the pixels near the large jump of brightness, for example on the contours of the objects. Temporal masking retains the local masking effect for a while, even if it is not situated in the next frame. The final thresholds appear as a result of enough complex and yet unknown interaction between all these processes.

The degree of visual independence achieved can be assessed as some consumption of bits per pixel (BPP) used for the individual components in comparison with total BPP of the source image.

Let $\mathbf{U} = \psi(c_0,c_1,...,c_M) = c_0 \oplus c_1 \oplus ... \oplus c_M$ is some certain composition of components c_0 ,c_1,...,c_M and $h(c_0)$, $h(c_1)$,...,$h(c_M)$ are corresponding thresholds of HVS. If the range of components is R, then these components can be quantized on $R/h(c_0)=N(c_0)$, $R/h(c_1)=N(c_1)$,...,$R/h(c_M)=N(c_M)$ levels. Let us assume that $N(c_0) <= N(c_1) <=...<= N(c_M)$ and corresponding numbars of BPP are $B(c_0) <= B(c_1) <=...<= B(c_M)$. Then the quantization of the whole composition will demand more long description than separate quantization of the components, because we have to use maximum number of levels $N_{max} = max(N(c_m))$, m = 0,...,M, to be sure in invisibility of quantization errors for the whole image, i.e.:

$$\sum_m \log N(c_m) \le \sum_m \log N_{\max} \Rightarrow \sum_m B(c_m) \le B(\Psi(c_0,c_1,...,c_M)). \qquad (3.2)$$

Existence of threshold properties of human vision, enables us to consider this feature as a kind of a compressor. The space of images is divided onto the multidimensional cells of complex shape. Within the cells images do not differ from each other. At the same time any existing compressor do the same, that is creates analogous set of cells. So the fact that we see some impairments on the decompressed image can be explained by the differences between the human's and compressor's cells. So, each human vision model may be considered as a compressor, and in contrast, any compressor is a sort of a vision model.

The subthreshold properties of a human vision are well investigated both in still and moving environment. That is why the compression task can be solved by reconciling threshold properties of compressor with thresholds of vision [32]. Besides of physical limitations of eye as itself, there is a lot of uncontrolled and subjective factors leading to large variety of estimates. Some users prefer "soft", slightly smoothed images (methods basing on the orthogonal transformations), other ones prefer more "rigid" with contrast contours (methods basing on the block quantization). Nevertheless, this task can be formalized under the object – oriented approach. Here the main task is to formalize an expert's requirements (for example, in terms of probability of false accept and false reject).

So, we consider the ideal compressor as one with some threshold properties, reconciled with the vision properties of the receiver. Among all such compressors some one, having best statistical properties of quantization cells, will ensure maximal compression.

3.1.3. Problem statement.

On Fig.6. typical scheme of a compressor and decompressor is presented. An image is decomposed by linear transformation to several components followed by quantization and lossless Huffman encoding. On the receiver side Huffman decoding and dequantization are performed, and finally the reverse liner transform restores decompressed image.

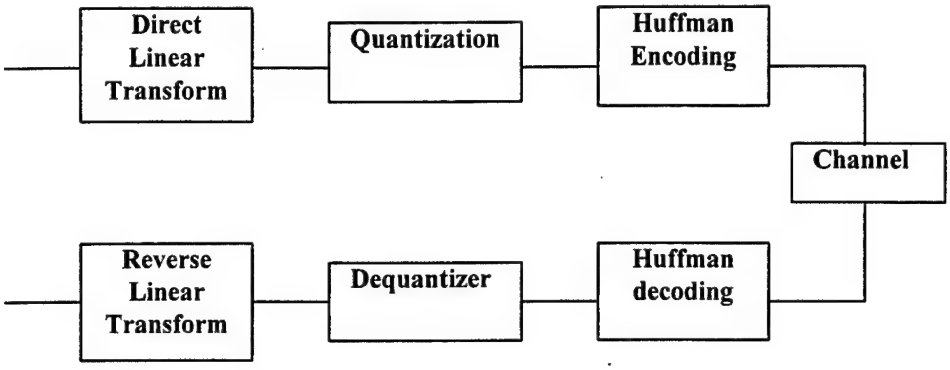


Fig.6. Typical flow chart of compressor and decompressor.

Let us assume that a source image consists of three major components: smoothed component $\mathbf{U_S}$, contour one $\mathbf{U_C}$, and residual, noise like $\mathbf{U_N}$ one:

$$\mathbf{U} = \mathbf{U_L} + \mathbf{U_H} = \mathbf{U_S} + (\mathbf{U_C} + \mathbf{U_N}). \qquad (3.3)$$

In most cases the component $\mathbf{U_N}$ does not carry any useful information, but takes rather large amount of bits to describe it. In contrast, $\mathbf{U_L}$ and $\mathbf{U_C}$ components usually contains much more information about the shape of the objects. That is why ideally we would like to have resulting image as sum of separated $\mathbf{U_L}$ and $\mathbf{U_C}$. It is relatively simple to separate $\mathbf{U_H} = \mathbf{U_S}$ component from the sum $\mathbf{U_H} = \mathbf{U_S} + \mathbf{U_N}$ using linear decomposition $\mathbf{W}$. It is because the $\mathbf{U_L}$ and $\mathbf{U_H}$ components are in different frequency area. In contrast, it is not trivial to separate $\mathbf{U_C}$ and $\mathbf{U_N}$.

To separate them, more complex, in general case, nonlinear filtering is needed. Primitive nonlinear operation – quantization is usually used in the most compression schemes. It is tuned onto the components with high energy, and removes others with low energy. This is in contradiction to HVS properties. We can clearly see low energy $\mathbf{U_C}$ component since it is highly structured, in contrast to $\mathbf{U_N}$ component which looks like random noise. Simple lowering of the number of quantization levels leads to fast reducing of the quality of decompressed image. From an expert's point of view, when a power of compression increases, it would be more reasonable to loose some other objects instead of those of interest. To solve this task it would be useful to separate useful objects from others and redistribute bits budget so, to reproduce them perfectly. In general, such a decomposition should rely on a priory information about the objects of interest. Here again we can consider the decompositions to line of images with different amount of a priory information, from simple contour detection to highly "intellectual", recognizing the objects. The scheme of a compressor with nonlinear decomposition is shown on Fig.7.

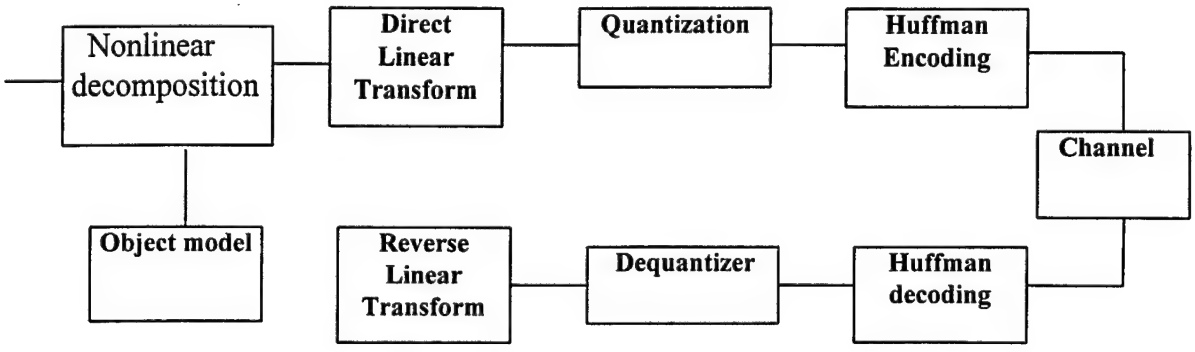


Fig.7. Compression with nonlinear decomposition.

The goal may be formulated in the following way. To investigate potential reserve of rate/distortion function of a compressor improving on the basis of chosen linear transformation, and simple nonlinear image decomposition separating $\mathbf{U_C}$ and $\mathbf{U_N}$ components.

3.2. Wavelet decomposition.

3.2.1. Wavelet basis

One of the most commonly used approaches for analyzing a signal $\mathbf{U(x)}$ is to represent it as a weighted sum of simple building blocks, called basis functions:

$$\mathbf{U(x)} = \sum_n \mathbf{C}_n \Psi(\mathbf{x}), \tag{3.4}$$

where $\psi_n(\mathbf{x})$ are the basis functions, and $\mathbf{c_n}$ are coefficients, or weights. Since the basis functions are fixed, the information about the signal contains only in the coefficients. There are many ways to choose basis functions. The simplest one uses a set of translated pulses as the basis, and it leads to spatial representation of the data. Choosing the set of harmonic curves as the basis functions, results in Fouirier representation, containing information in frequency domain [28-31].

For the signal compression purposes, neither of the above representations is ideal. We would like to have a representation, which contains information about both the space and frequency behavior of the signal. We want to know the frequency content of the signal only at a particular instant in space. However, spatial resolution (Δx) and frequency resolution ($\Delta\omega$) cannot be chosen enough small together at the same time, because their product is bounded below by the Heisenberg inequality:

$$\Delta x \Delta\omega \geq ½. \tag{3.5}$$

This inequality means that we must trade off time resolution for frequency resolution. Thus, it is possible to have very good spatial resolution if you are willing to settle low resolution in frequency, or, vice versa, it is possible to have very good resolution in frequency if you are willing to settle low spatial resolution.

By their nature, low frequency events are spread out (or non-local) in space, and high frequency events are concentrated (or localized) in space. Thus, to left within the confines of the Heisenberg inequality and to have sufficient spatial-frequency information about a data, we have to construct the basis functions like cascaded octave filters, which repeatedly split the data's bandwidth in a half.

Let us constrain all of the basis functions in $\{\psi_n\}$ to be scaled and translated versions of the same prototype function ψ, known as the mother wavelet. The scaling is accomplished by multiplying $\mathbf{x}$ by some scale factor. If we choose the scale factor to be a power of 2, yielding $\psi(2^\nu \mathbf{x})$, where ν is some integer, we get the desirable cascaded octave bandpass filter structure. Because the function ψ has finite support, it will need to be translated along the space axis in order to cover an entire data. This translation is accomplished by considering all the integral shifts of ψ:

$$\psi(2^\nu \mathbf{x} - \mathrm{k}), \qquad \mathrm{k} \in \boldsymbol{Z}. \tag{3.6}$$

This really means, that we are translating ψ in steps of size $2^{-\nu}\mathrm{k}$, because $\psi(2^\nu \mathbf{x} - \mathrm{k}) = \psi(2^\nu(\mathbf{x} - 2^{-\nu}\mathrm{k}))$. Putting this all together, gives us a wavelet decomposition of the signal

$$\mathbf{U(x)} = \sum_\nu \sum_k \mathbf{C}_{\nu k} \Psi_{\nu k}(\mathbf{x}), \tag{3.7}$$

where $\psi_{\nu k}(\mathbf{x}) = 2^{\nu/2}\psi(2^\nu \mathbf{x} - \mathrm{k})$ (the multiplication by $2^{\nu/2}$ is needed to be the basis orthonormalized). The coefficients $\mathbf{c}_{\nu k}$ must be computed by wavelet transform, which is just the inner product of the signal $\mathbf{U(x)}$ by corresponding basis function $\psi_{\nu k}(\mathbf{x})$. Wavelet may be implemented as octave bandpass filters, which in turn may be represented as cascaded implementation of "lowpass" $\{h_\mathrm{k}\}$ and "highpass" $\{g_\mathrm{k}\}$ filters. Below we briefly describe the implementation details of the wavelet-based image compression.

3.2.2. Implementation of the wavelet transformation.

At the beginning the compressed signal is decomposed to the pair of components with low and high frequencies using the convolution-subsampling operators. It is performed in the spatial domain with the helps of "lowpass" $\{\boldsymbol{h}_\mathbf{k}\}$ and "highpass" $\{\boldsymbol{g}_\mathbf{k}\}$ filters. Let $\boldsymbol{H}$ and $\boldsymbol{G}$ be such convolution–subsampling

operators realizing these filters, and H^* and G^* be their adjacent (i.e. upsamling–anticonvolution) operators. We may choose filters of finite length (L) and satisfy the following conditions of orthogonality and perfect reconstruction:

$$\mathbf{HG^* = GH^* = 0}, \qquad \mathbf{H^*H + G^*G = I}, \tag{3.8}$$

where I is the identity operator.

This decomposition process is iterated on the low frequency components. Let $\mathbf{U}=\{\mathbf{u_k}\}^{N-1}_{k=0}$ be a vector to be decomposed. Then, the convolution-subsampling operations transform the vector **U** into two subsequences each of N/2 elements length: $H\mathbf{U}$ and $G\mathbf{U}$. Next, the same operations are applied to the vector $H\mathbf{U}$ to obtain $H^2\mathbf{U}$ and $GH\mathbf{U}$ of length N/4. If the process is iterated J times, we have the discrete wavelet coefficients ($G\mathbf{U}$, $GH\mathbf{U}$, $GH^2\mathbf{U}$, ... ,$GH^J\mathbf{U}$, $H^{J+1}\mathbf{U}$) of length N. As a result, the wavelet transform analyzes the data by partitioning its frequency content dyadically finer and finer toward the low frequency region (i.e., coarser and coarser in the original space domains).

The reconstruction (or synthesis) process is also very simple: starting from the lowest frequency components (or coarsest scale coefficients) $H^{J+1}\mathbf{U}$, and the next frequency components $GH^J\mathbf{U}$, the adjacent operations are applied and added to obtain $H^J\mathbf{U} = H^*H^{J+1}\mathbf{U} + G^*GH^J\mathbf{U}$. This process is iterated to reconstruct the original vector **U**. The computational complexity of the decomposition and reconstruction process is in both cases O(N) as easy to see.

We use the following anti-aliasing relations between analysis $\{h_k\}$, $\{g_k\}$ and synthesis $\{h^*_k\}$, $\{g^*_k\}$ filters:

$$h^*_k = (-1)^n g_{k-1}, \qquad g^*_k = (-1)^{n-1} h_{k-1} \tag{3.9}$$

The filters coefficient is presented in Table 4.

Table 4.

Tap	*Approximate value*
h_0	0.85269867900940
$h_1 = h_{-1}$	0.37740285561265
$h_2 = h_{-2}$	-0.11062440441842
$h_3 = h_{-3}$	-0.0238494650119380
$h_4 = h_{-4}$	0.037828455506995
g_{-1}	0.78848561640566
$g_{-2} = g_0$	-0.41809227322221
$g_{-3} = g_1$	-0.040689417609558
$g_{-4} = g_2$	0.0645388826282938

3.2.3. Source image

We assume, that source images are captured with the precision of 8 bits/pixel. Before the encoder will compute the discrete wavelet transform of the image, the samples **U(x,y)** must be transformed according to the following equation:

$$\mathbf{U^*(x,y) = (U(x,y) - M)/R}, \tag{3.10}$$

where **M** – mean value of the image pixels, but $\mathbf{R = \max (U_{max} - M, M - U_{min})/128}$, $\mathbf{U_{min}}$ and $\mathbf{U_{max}}$ are, respectively, the minimum and maximum pixel values in the image.

3.3. Quantization.

3.3.1. Subband variance computation

A subband variance estimate is calculated basing on a subregion of each subband. Let $a_k(x,y)$ denote the floating point array of width X_k and height Y_k, comprising the k^{th} subband. The width and height of the subregion to be used for the estimate, are:

$X_k^* = \text{round}(3X_k/4)$
$Y_k^* = \text{round}(7Y_k/16).$

The function round(...) denotes rounding of a floating point value to the nearest integer. The variance will be computed by the unbiased estimator:

$$\sigma^2 = \frac{1}{\mathbf{X}_k^* \mathbf{Y}_k^* - 1} \sum_{n=x_{D,k}}^{x_{1,k}} \sum_{m=y_{D,k}}^{y_{1,k}} (\mathbf{a}_k(m,n) - \mu_k)^2, \tag{3.11}$$

where μ_k denotes the mean value of a_k. The horizontal and vertical offsets for the subregion ($x_{p,k}$ and $y_{p,k}$), relative to the upper left corner, will be:

$$x_{0,k} = |X_k/8|, \quad x_{1,k} = x_{0,k} + X_k^* - 1,$$
$$y_{0,k} = |9Y_k/32|, \quad y_{1,k} = y_{0,k} + Y_k^* - 1. \tag{3.12}$$

3.3.2. Bin width calculation.

The formula for the bin widths Q_k is:

$$Q_k = \begin{cases} 1/q, & \text{if } k = 0, \\ \dfrac{10}{q^* \log_e(\sigma_k^2)}, & \text{otherwise} \end{cases} \tag{3.13}$$

The zero bin Z_k is computed as $Z_k = 1.2 \cdot Q_k$.

3.3.3. Quantization procedure.

Each subband is coded separately according to a scalar quantizer characteristic having uniform width bins with the exception of the zero bin, which is 20% wider. The quantization transforms a floating-point wavelet coefficient C to an integer quantizer index q, that indicates the quantizer bin in which C lies. The dequantization transforms the index q to a real number C_q, representing all data values that lie within that bin. Encoder transmits the values of Q_k (the bin width) and Z_k (the zero bin width) for each subband along with the Huffman coded quantizer indices. Quantization of the k^{th} two-dimensional subband, $C_k(m,n)$, is given by the following:

$$\begin{aligned} &\text{if } (C_k(m,n) > Z_k/2), && q_k(m,n) = \lfloor (a_k(m,n) - Z_k/2)/Q_k \rfloor + 1; \\ &\text{if } (-Z_k/2 \le C_k(m,n) \le Z_k/2), && q_k(m,n) = 0; \\ &\text{if } (C_k(m,n) > Z_k/2, && q_k(m,n) = \lceil (a_k(m,n) + Z_k/2)/Q_k \rceil - 1. \end{aligned} \tag{3.14}$$

The notation $\lceil . \rceil$ and $\lfloor . \rfloor$ denotes the functions those round values to the next largest and next lowest integer, respectively. The quantified wavelet coefficients produced by the dequantizer, are given by:

$$\begin{aligned} &\text{if } (q_k(m,n) > 0), && C_k^*(m,n) = (q_k(m,n) - B)Q_k + Z_k/2; \\ &\text{if } (q_k(m,n) = 0), && C_k^*(m,n) = 0; \\ &\text{if } (C_k(m,n) > Z_k/2, && C_k^*(m,n) = (q_k(m,n) + B)Q_k - Z_k/2, \end{aligned} \tag{3.15}$$

where B is some parameter between 0 and 1 that determines the reconstructed values. Note, that if B = ½ then the reconstructed value, corresponding to each quantization bin, would be the bin's midpoint. We used B = 0.44. The quantizer indices $q_k(m,n)$ are transmitted losslessly by a combination of zero run-length and Huffman encoding.

3.3.4. Entropy encoding

A Huffman encoder is used to assign variable length codes to the quantized coefficients within a block. Special codes are provided for zero runs. Coefficients and zero run lengths outside the range of the table are embedded in the code stream with an escape sequence. Table 2. lists the complete symbol set

Table 5. Huffman table of input symbols.

Position	Value
1	zero run length 1
2	zero run length 2
3	zero run length 3 .

•
•
•

100 zero run length 100
101 escape for position 8 bit coefficient
102 escape for negative 8 bit coefficient
103 escape for position 16 bit coefficient
104 escape for negative 16 bit coefficient
105 escape for zero run 8 bits
106 escape for zero run 16 bit
107 coefficient value –73
108 coefficient value –72
109 coefficient value –71

•
•
•

180 - use position 1 only –

•
•
•

253 coefficient value 73
254 coefficient value 74

As we are interested in gain in r/d characteristic, we do not implement real Huffman encoding and use only the value of pseudo entropy H, calculated as:

$$H_{pseudo} = -\sum_{m} P(m)\log_2 P(m), \tag{3.16}$$

where $P(...)$ is a histogram of the symbols, obtained in accordingly to some accepted standard, before the Huffman encoding.

3.4. Nonlinear decomposition

Acceptable nonlinear decomposition, separating an image to components $\mathbf{U_C}$ and $\mathbf{U_N}$, musts rely on the vision properties of the receiver. Our ability to detect, to discriminate, and to process week signals with a noise is based on a priory knowledge of signal's shape. During the long evolution, our vision system produced a number of special receptive field, capable quickly respond to the known signals. It is well known, that the longer and bigger is an object, the higher is the response of human vision to them. Small spot-like objects are hardly visible even on the complex background. It is because the sensitivity of human vision to such objects is high. It can be partially explained by the mechanism of signal summation. If we know the shape of a signal, we can use some special filter, matched to this signal, and detect the places where this filter has large response. It is known, that human vision has multiple specialized receptive fields replying only on the structured stimulus, as lines differently oriented in space [33].

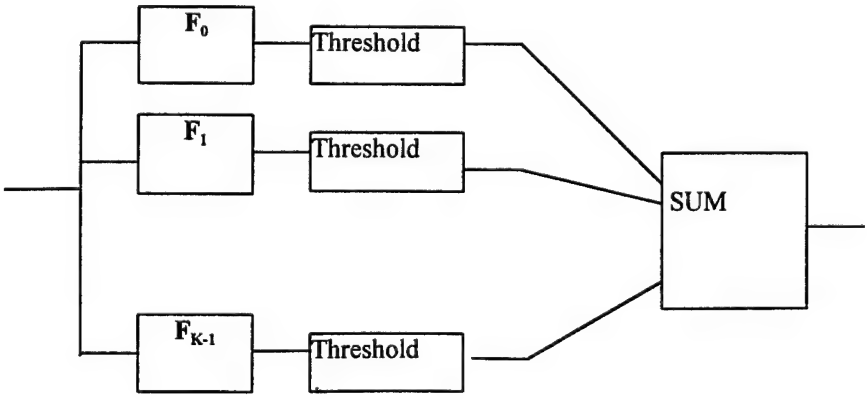


Fig.7.

The response of these receptive fields can be modeled by the convolution of the matrix of weighting coefficients with an image, i.e.:

$$\mathbf{R} = \mathbf{U} \otimes \mathbf{F}, \tag{3.17}$$

where **F** is local matrix modeling the receptive field, symbol ⊗ means convolution, and **R** is a response of the receptive field on a signal **U**.

We will not discuss this question further, but will describe the simple model of the receptive field. Let there are N simple receptive fields, tuned on the special structured signal R_k (k = 0,…,K-1). Then the behavior of the complex field can be described as the sum of the replies of simple fields after passing the threshold operation. On Fig.7. we can see such a structure.

We used the following simplest receptive fields:

Horizontal direction *Vertical direction*

$\mathbf{F_0}$	$\mathbf{F_1}$	$\mathbf{F_2}$	$\mathbf{F_3}$
0 0 0	-1 -1 -1	-1 1 0	0 1 -1
1 1 1	1 1 1	-1 1 0	0 1 -1
-1 -1 -1	0 0 0	-1 1 0	0 1 -1

Diagonal direction

$\mathbf{F_4}$	$\mathbf{F_5}$	$\mathbf{F_6}$	$\mathbf{F_7}$
1 0 0	1 -1 -1	0 0 1	-1 -1 1
-1 1 0	0 1 -1	0 1 -1	-1 1 0
-1 -1 1	0 0 1	1 -1 -1	1 0 0

The reply of this fields is calculated by

$$\mathbf{R}_p(k,l) = \sum_{m=-1}^{1}\sum_{n=-1}^{1}\mathbf{U}(k+m,l+n)\cdot\mathbf{F}(m+1,n+1), \tag{3.18}$$

$p = 0,\ldots,7.$

$$\mathbf{R}_p^{Th}(k,l) = \begin{cases} 0, & if \quad \mathbf{R}_p(k,l) < T \\ \mathbf{R}_p(k,l), & if \quad \mathbf{R}_p(k,l) \geq T \end{cases} \tag{3.19}$$

Resulting reply **R** is the average of partial replies $\mathbf{R}_p{}^{Th}(k,l)$:

$$\mathbf{R}(k,l) = \frac{1}{8}\sum_p \mathbf{R}_p^{Th}(k,l). \tag{3.20}$$

So, as a model of $\mathbf{U_C}$ contour component we use (3.20). For smoothed $\mathbf{U_S}$ component we can use local mean, that is $\mathbf{U_S} = \mathbf{M}$:

$$\mathbf{M}(k,l) = \frac{1}{9}\sum_{m=-1}^{1}\sum_{n=-1}^{1}\mathbf{U}(k+m,l+n). \tag{3.21}$$

It is evident that if threshold T is zero, then (3.20) is simple linear laplacian $\mathbf{R}_L$, that is

$$\mathbf{R}_L(k,l) = \frac{1}{8}\sum_p \mathbf{R}_p(k,l) = \mathbf{U}(k,l) - \mathbf{M}(k,l). \tag{3.22}$$

On the borders for indices k and l we use the following rule:

k= min (max (0, k+m),K),

l = min (max (0, l+n),L),

where K and L are the numbers of lines and columns in an image.

As was mentioned above, the nonlinear decomposition can be applied as before as after the linear decomposition. For the linear case we will use (3.20), but for the nonlinear one, the following combined nonlinear filter will be used:

$$\mathbf{R}_{\mathbf{S+C}}(k,l) = \mathbf{M}(k,l) + \mathbf{R}(k,l), \tag{3.23}$$

that is, if the reply **R**(...) equals zero, the signal value is substituted by the local mean value.

3.5. Experimental results.

The goal of the experiment was to reveal further reserves of rate/distortion ratio of wavelet-based strategy consisting in the quantizing and the entropy encoding of wavelet coefficients.

To assess resulting images we will use subjective estimation in a fixed viewing conditions and well-known formula for calculating of PSNR:

$$\mathbf{PSNR} = 20\log_{10}\frac{255}{\sqrt{\mathbf{MSE}}}, \tag{3.24}$$

where

$$\mathbf{MSE} = \frac{1}{M \cdot N}\sum_{m}\sum_{n}(\mathbf{U}(m,n) - \mathbf{U}^{*}(m,n))^2, \tag{3.25}$$

and M and N are the number of lines and columns in the image.

3.5.1. Components

The low-pass component $\mathbf{U_S}$, obtained on the original image, corresponds to linear decomposition. It may be shown, that component $\mathbf{U_H}$ contains both a mixture of contours $\mathbf{U_C}$ and noise $\mathbf{U_N}$ components. Applying linear decomposition to the component $\mathbf{U_H}$ one can not separate needed components. Really

$$F\mathbf{U_H} = F(\mathbf{U_C} + \mathbf{U_N}) = F\mathbf{U_C} + F\mathbf{U_N} = F\mathbf{U_C}, \tag{3.26}$$

where $F\mathbf{U_N}=\mathbf{0}$, but $F\mathbf{U_C}$ is smoothed contour component. Adding of this component to the smoothed $\mathbf{U_S}$ results an image with reduced resolution. In turn, noise-like component $\mathbf{U_H} - F\mathbf{U_H}$ contains considerable part of contour component. It means, that to separate those components perfectly we need to use nonlinear decomposition.

If we use nonlinear decomposition, we managed to obtain good contour component $\mathbf{U_C}$. As a result we have perfect restored image $\mathbf{U_S} + \mathbf{U_C}$. It is seen also, that component $\mathbf{U_N}$ looks really like noise with negligible part of contour component.

Qualitative estimate demonstrates, that nonlinear decomposition gives 36.823 dB, but linear one – only 27.315 dB.

3.5.2. Rate/distortion properties.

The rate/distortion results are presented in Table 6. For comparison, several images were compressed by standard wavelet technology and proposed one with different quantization steps.

Table 6. *Standard compressor* *Proposed compressor*

	Bits/pixel	*PSNR*	*Bits/pixel*	*PSNR*
Q = 10	1.638	34.22	1.50	33.98
Q = 20	1.11	30.9	1.05	30.2
Q = 30	0.85	29.26	0.8	28.9
Q = 40	0.725	28.16	0.67	28.3

It was demonstrated, that nonlinear decomposition gives us perfect separating of contour and noise component. In contrast, linear decomposition with adaptive quantization of the subbands gives less value

of a gain. Wavelet compression with nonlinear decomposition can be useful in air and space images, where preserving of small object corrupted by noise may be important.

4. CONCLUSION

An image was considered as a blend of several statistically and semantically independent components, each of them containing details of different information classes and statistical properties. In the above discussion we pursued the goal to decompose an image to two components: smoothed and texture ones. The tasks of image enhancement, restoration, and compression were considered from the position of preliminary decomposition and following manipulating with obtained components.

Image decomposition helps to decide many different tasks, like noise removing, enhancement of gray-scale, color, and multispectral images, extraction of texture, noise-free detection of contours, localization of image details, segmentation, and some others. We proposed rank algorithms for image decomposition and image enhancement, using statistics for local window. The efficiency of algorithms was verified by processing of real and artificial images.

Real image is enough complex signal, and its simplification by decomposing onto several information components looks to be perspective approach to the decision of different image processing and analysis tasks. Separating from an image only the component of interest, we may get some new starting point for the decision of many traditional tasks, the set of which is significantly wider than it was discussed above. For example, extracting texture component and avoiding average-brightness information, one can decide texture image segmentation more efficiently. Varying area parameters for relating objects to one or another component, one can detect objects from an image regarding their sizes, etc.

5. REFERENCES

1. W.K.Pratt, *Digital Image Processing*. New York: Wiley, 1978.
2. A.G.Vitoushkin *On the Multidimensional Variations.* Moscow, Gostekhizdat, 1955 (in Russian).
3. *Proceedings of symposium on image modeling. Rosemont, Illinois, 1979.* Computer Graphics and Image Processing. 1980, V.12 (Special issue on image models.), no.1-4, 426 p.
4. A.N.Tikhonov, V.J.Arsenin *Methods of Solving Ill-Posed Problems.* Moscow, Nauka, 1979 (in Russian).
5. O.P.Milukova *Digital Image Restoration.* Moscow, IPPI, 1988 (in Russian).
6. O.P.Milukova "Fourier Transform in Restoration Problem" Proc. SPIE, v.2363, 1995, pp.98-103.
7. L.D.Landau, Ye.M.Lifshits "Field Theory", Moscow, Nauka, 1967 (in Russian).
8. G.A.Mastin, "Adaptive filters for digital image noise smoothing: An evaluation.," Computer Vision Graphics Image Process. Vol.31, 1985, no.1, pp.103-121.
9. P.A. Chochia "Image Enhancement Using Sliding Histograms," Computer Vision, Graphics, and Image Processing, V.44, 1988, no.2, P.211-229.
10. P.Perona, J.Malik "Scale-Space and Edge Detection Using Anisotropic Diffusion," IEEE Trans. PAMI, 1990, Vol.12, no.7, pp.629-639.
11. S.Nishikawa, R.J.Massa, J.S.Mott-Smith "Area Properties of Television Pictures," IEEE Trans. IT-11, 1965, no.3, pp.348-352.
12. P.A.Chochia *Image processing and analysis on the basis of two-scale image model.* Moscow, IPPI, 1986 (in Russian).
13. H.A.David *Order Statistics.* New York: Wiley, 1970.
14. J.S.Lee, "Digital image smoothing and the sigma filter," Computer Vision, Graphics, Image Process., Vol.24, 1983, pp.255-269.
15. A.Rozenfeld *Picture processing by computers.* New York, Academic Press, 1969.
16. I.Scollar, B.Weidner and T.S.Huang, "Image enhancement using the median and the interquartile distance", Computer. Vision Graphics Image Process. 25, 1984, no.2, pp.236-251.

17. R.Wallis, "An approach to the space-variant restoration and enhancement of images", in *Image Science Mathematic* (C.O.Wilde and E.Barrett, Eds); Proc. Sympos. Current Mathem. Problems in Image Science, Monterey, Calif., Nov.1976. Western Periodicals, North Hollywood, 1977, pp. 10-12.
18. T.S.Huang, "Image Enhancement: A Review," *Opto-Electronic*, Vol.1, pp.49-59, 1969.
19. A.Rosenfeld and A.C.Kak, Digital Picture Processing, Academic press, New York, 1976.
20. E.L.Hall, "Almost Uniform Distribution for Computer Image enhancement," *IEEE Trans. Comput.* Vol.C-23, no.2, pp.207-208, 1974.
21. R.A.Hummel, "Image Enhancement by Histogram Transformation," *Computer Graphics, and Image Processing*, Vol.6, no.2, pp.184-195, 1977.
22. Fu-Nian Ku, "The Principles and Methods of Histogram Modification Adapted for Visual Perception," *Computer Vision, Graphics, and Image Proces.*, Vol.26, no.1, pp.107-117, 1984.
23. S.K.Mitra and Tian-Hu Yu, "Transform Amplitude Sharpening: A New Method of Image Enhancement," *Computer Vision, Graphics, and Image Processing*, Vol.40, no.2, pp.205-218, 1987.
24. P.A.Chochia, "The Two-Components Image Model," *Image Coding and Processing*, p.69-87, Nauka, Moscow, 1988. (in Russian).
25. P.A.Chochia, "Image Processing Methods on the Basis of Two-Components Image Model", *Image Coding and Processing*, p.98-111, Nauka, Moscow, 1988. (in Russian).
26. J.L.Marroquin, "Random Measure Fields and the Integration of Visual Information." *IEEE Trans. Systems, Man and Cybernetics*, Vol.22, No. 4, pp. 705-716, 1992.
27. Chun-Hsien Chou and Yun-Chin Li, "A perceptually Tuned Subband Image Coder Based on the Measure of Just-Noticeable-Distortion Profile," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.5., No.6., November 1995, pp.467-475.
28. I.Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Match.*, vol.XLI, pp.909-996, 1988.
29. A.Laine, J.Fan, *An adaptive Approach for Texture Segmentation by Multi-Channel Wavelet Frames*, Center for Computer Vision and Visualization, August, 1993, TR-93-025.
30. M.Acheroy, S.Grandjean, *METEOSAT Image Compression using the Wavelet Transform*, Final Report, March 14, 1994.
31. M.V.Wickerhauser, *Lectures on Wavelet Packet Algorithms*, Department of Mathematics, Washington University, St, Louis, Missouri 63130, Nov. 18, 1991.
32. A.B.Watson, G.Y.Yang, J.A.Solomon, and J.Villasenor, "Visual thresholds for wavelet quantization error in Human Vision and Electronic Imaging," B. Rogowitz and J.Allebach Ed., *Proceedings of the SPIE 2657*, pp. 382-392.
33. A.B.Watson, *Digital images and human vision*, Cambridge, MA. MIT Press, 1993.
34. A.B.Watson, G.Y.Yang, J.A.Solomon, J.Villasenor, *Visual thresholds for wavelet quantiztion error*, NASA Ames Research Center, Moffett Field, CA, 94035-1000, Department of Electrical Engineering, UCLA, Los Angeles, CA 90024-1594.
35. R.L.Stratonovitch, *Information theory*, Moscow, "Sov. Radio", 1975, 424 p.